

# コンピュータガイド

－ インターネット編 －

－cc 環境の利用者へ－

京都産業大学  
計算機センター教育研究システム課

初版'94. 3.23

改訂'99. 4. 8

## Kyoto Sangyo University Style of Computing

1998年12月15日、新聞各紙の「京都産業大学 教育にLinuxを採用」の報道に、本学が教育機関のみならず、企業の情報担当者からも注目をうけることになったのはすでに御承知かもしれません。

これまでLinuxは、研究機関や大学の研究室に徐々に導入され、Open Sourceとしてその有効性が認められつつありましたが、教育環境への大量で本格的な採用における前例と実績たるや皆無に等しく、大学が真剣に取り組むべき情報教育を支える設備に、Linuxを試験的採用ではなく実用教育設備として大規模導入を決断した本学に、日本のみならず世界から注目を受けることは当然といえましょう。

この背景には、現代の情報化はMicrosoft社の技術に依存し過ぎていたことがあげられます。これは、情報化にあたり企業においてすら、Microsoft以外の選択肢を採用することが困難な状況がここしばらく続いている現状を指しています。

最近になって、同社の提供する技術だけで情報化を進めることに、少なからず疑問を抱く大学や企業の情報担当者が国の内外を問わず増えつつあり、Microsoft社のそれと対等、あるいはそれ以上に競えるまったく新しい選択肢を多くの人たちが待ち望んでいるそのさなかでのこの報道が、如何に衝撃を持って情報担当者に伝わったかは、想像に難くありません。

本学と取り引きのある企業の方から、営業先の大学や企業を訪れる度に、必ずといっていいほど京都産業大学とLinuxの話題になるときが多くなっており、まさに「渦中の」といった感があります。

しかし、本学にとってLinux採用は特別な選択ではなく、むしろ何の疑いもはさむことのない自然な決定であることは、本学の情報化の歴史をほんの少し振り返って見れば理解できることなのです。

その歴史に少しだけ触れてみましょう。

本学は1965年の開学当初から、コンピュータの実用化と情報教育に極めて積極的に取り組んできました。

2年目の1966年には、計算機科学研究所が理事会直属として設置されました。理事会直属とは、大学の下にはなく、すべての学部を横断して正しい方向に情報化を推進するため、あえて大学と等しい位置づけで設置されたことを意味します。

さらに当時は国立大学の大学院生ですらコンピュータにはなかなか触ることができないのが実情でしたが、本学は、すべての学部学生を対象にコンピュータ実習教育を開始しています。このようなことは前例がなく、もちろん日本で最初の取り組みとなりました。

さらに、3年目の1967年には図書洋書目録がコンピュータ化され、これはおそらく世界初のシステムでありました。その後、続けさまに入試業務や教務課受講登録、さらに成績管理システムなどが日本初という輝かしい歴史を刻みながら学内で開発されました。

ことに1969年の入試におけるマークシート方式は、近年ではセンター試験をはじめどこでも行なわれていますが、日本で最初にマークシート方式による入学試験を実用化したのは、他ならぬ本学であったことを明記すべきでありましょう。

さらに、情報教育設備に国庫助成が適用されたころ、この時、国には日本の情報産業会を育てようとの意図があり、各大学はこぞって国産コンピュータの導入を進めるなか、本学はまるで流れに逆らうように、米国からTSS(タイムシェアリング:時分割)処理に優れたコンピュータ(DEC SYSTEM-20)を1976年に導入しています。MIT(マサチューセッツ工科大学)で開発されたOSを持つこのコンピュータのTSS性能と使い勝手の良さは郡を抜いており、これによって全学部の学生に直接コンピュータを操作させ、情報教育に大きな成果をあげることができました。

また、本学のネットワークKINGを例にとっても、敷設の1987年当時は大型汎用機全盛期であり、多くの大学が汎用機メーカーのネットワークを採用するなか、本学はもちろん汎用機も持っていました。学内のすべてのコンピュータの異機種間相互接続を優先して、あえて処理能力の低いEthernetを選択し導入しています。

汎用機メーカーのネットワークはずいぶん昔にその姿を消しましたが、Ethernet が今日どのような位置付けにあるかをみれば、その時の選択の良否はあえてここで述べる必要はないでしょう。この KING ネットワークは、KING2 とも呼ぶべき 1995 年の FDDI スイッチド・ネットワーク導入へと継承され、このスイッチは昨年まで学内ネットワークの中心設備として十分な性能を発揮し、そしてこの 1999 年、KING3 とも呼称すべき GIGABIT Ethernet スイッチ導入へと順当ともいえる成長を遂げつつあることは、本学の 2 年生以上の学生なら理解できるでしょう。

ところで、本学が FDDI スイッチを採用した 1995 年という年は、国によって大学ネットワーク整備が強力に推進され、国公立大学に ATM ネットワークが競うように導入された時期でありました。しかし、この時も本学は ATM の有効性への懸念と確実に稼働させることへの技術的困難さを理由に採用せず、ATM より明らかに性能が低いと見做された FDDI スイッチを採用するに至っています。ところが、国によるその後の調査で、当時多くの大学に導入された ATM ネットワークは、ほとんど満足に稼働できていないことが明らかとなっています。

特記すべきは、KING から KING3 にいたるネットワークの導入にあたり、設計を業者に頼らず、すべて学内で行ったという事実です。

このように特異ともみえる情報化の歴史をあげれば枚挙にいとまがありませんが、本学にはその健学の当初から培われたコンピューティング・スタイルが存在することに気づかれるでしょう。

- 情報化への独自の判断と見解を持ち、いつ如何なる条件であっても本学にとって将来を見据えた最善の決定をしようとしたこと
- そのような決定を導くべき、飽くなきチャレンジ精神が存在すること
- その決定は時として独創的であるがゆえに他に例がなく、時として本学は最前線（フロンティア）にいること
- 本学における最良の策が必ずしも他におけるそれとはならないように、他がどのような情報化を進めようとしているかを真似ようとはしないこと

これらは決して誇張でも、こじつけでもないことは、先に述べた情報化への歴史の一端を振り返るだけで、どなたにでも理解いただけるでしょう。

Linux 採用が、本学にとっては教育的に自然な解であり、必然であったことは、昨年 6 月頃の Linux 導入議論が、検討委員会のみならず全学部教授会をあげてわずか一ヶ月ほどで意志の統一をみたことから明らかであります。このことをとって、たとえ社会から本学の Linux 採用がリスクを伴う決定とみえても、本学にはそれをものともしない独創性とチャレンジ精神が既に存在していることは疑うべきもありません。

健学当初から現在に至る、この情報化への独創性とチャレンジ精神をして、Kyoto Sangyo University Style of Computing と誇らかに呼ぼうではありませんか。

京都産業大学に学ぶ学生諸君は、この伝統あるキャンパスで学んでいるわけですから、まさに選ばれた人だと申せましょう。どうか誇りと自信を持って、胸を張って大学の街京都を湯歩し、インターネットの世界においても堂々と振舞っていただきたいと希望します。

最後に紙面を借りてお願いがあります。

今春稼働する新しい教育情報システム cc 環境は、まだまだ不備を抱えています。

これをより使いやすい情報システムとして充実させてゆくのは、他ならぬ利用者です。

目先の利益を追うことなく、情報化とは本来どうあるべきかの前向きな議論をしていただきたい。本学には誰でも議論に参加できる sandai.\* ニュースグループが運用されております。この議論に参加することで、cc 環境は一層改善され、加えて参加者自身も大きく成長することが期待できます。

新しい教育情報システムの誕生を機に、皆さんの積極的な参加を願ってやみません。

計算機センター教育研究システム課

## コンピュータガイドの読み方

コンピュータガイドは、初心者および入門用として、電子メール、ネットニュースなどの利用方法を中心とした手引書です。一歩踏み込んで中級、上級者になろうとする方には本学 WWW ページ、<http://www.kyoto-su.ac.jp/information/index-j.html> に随時情報が追加されていく予定ですのでこれを参考に一般の書籍も交えて勉強すればいいでしょう。え？なんのことが判らない？大丈夫です。この本を読み終る頃には上で書かれている意味が判るようになっていきますから。

本編の「インターネットの手引」は、前者の初心者および入門用として記述しており、京都産業大学の共用 UNIX コンピュータを利用して生活する為の手引書として書かれてあります。1「インターネット利用に関するガイドライン」では全ての利用者に共通の注意を説明しています。2で基本的な用語や操作の説明、3で各情報処理教室からの使い方を説明、そして4から5まででUNIXの基本的な操作方法から始まり、電子メールやネットニュースなどが使えるようになるまでを説明しています。ここまでは全てのUNIX利用者に習得して欲しいと思います。コンピュータのことをほとんど何も知らなくても、何とかここまではたどりつけるように書いたつもりです。頑張ってください。

後の章は、UNIXのより進んだ使い方、自宅のコンピュータ(パソコン)から通信網(電話)を利用して本学のネットワークに接続する方法などについて説明しています。是非、頑張ってください。

また、巻末の付録にはリファレンスとして各種コマンドや記号などの一覧表、情報処理教室を利用する際の注意、著作権法の抜粋などがまとめられています。参考文献の一覧も付けてあります。

# 目次

第1章	インターネット利用に関するガイドライン	1
1.1	インターネットとは	1
1.1.1	インターネットで利用できるサービス	1
1.2	インターネットの利用手続	2
1.2.1	利用資格の取得	2
1.2.2	本学関係者への発行	2
1.2.3	学外者への発行	2
1.3	インターネットを使う上で知っておくべきこと	3
1.3.1	ネチケットガイドライン	3
1.3.2	セキュリティ	4
1.3.3	適切なパスワード	4
1.3.4	使ってはいけないパスワード	4
1.4	インターネットに関わるには	5
1.4.1	わからないことがあれば	5
1.4.2	電子メールやネットニュースの利用	6
1.4.3	WWWの利用	7
1.5	利用に関わる諸規則	8
1.5.1	遵守すべき事項	8
1.5.2	禁止される行為	8
1.5.3	規則違反に対する措置及び罰則	8
1.5.4	利用資格の消滅等	9
1.6	本学の責任免責および経費の負担	9
1.6.1	免責	9
1.6.2	経費の負担	9
1.7	著作権などの知的財産権について	10
1.8	情報処理教室の利用について	12
1.9	利用者サポート	13
1.9.1	メインカウンター	13
1.9.2	困ったときの補助員制度	13
第2章	みんなで使うコンピュータ	15
2.1	計算機センター運用の環境	15
2.1.1	cc環境	15
2.1.2	ユーザIDとパスワード	15
2.1.3	cc環境のUNIX	16
2.1.4	cc環境のWindowsNT	16
2.1.5	UNIXとWindowsNTのユーザID	16

2.1.6	情報処理教室のコンピュータ環境	17
2.1.7	電子メールアドレス	18
2.1.8	ユーザ ID の変更	18
2.1.9	パスワードの変更	19
2.1.10	パスワードの安全対策	19
2.2	login(ログイン) と logout(ログアウト)	22
2.3	キーボード	22
2.4	ウインドウ環境	24
2.4.1	マウス	24
2.4.2	ウインドウ環境の画面	25
2.4.3	メニュー	27
2.4.4	ボタン	28
2.4.5	ウインドウ環境のトラブル傾向と対策	28
2.5	どこから login?	30
2.5.1	パソコンから UNIX を利用する為の予備知識	30
2.6	コンピュータウイルス	32
<b>第 3 章</b>	<b>UNIX への道</b>	<b>33</b>
3.1	これ以降の読み方	33
3.2	Ultra1 を使う	34
3.2.1	まず部屋へ	34
3.2.2	電源を入れる	34
3.2.3	login する	35
3.2.4	ちょっとメッセージ	36
3.2.5	ターミナルはあるかな?	36
3.2.6	パスワードを変える	38
3.2.7	logout する (セッションを終了する)	40
3.2.8	シャットダウンする	40
3.2.9	部屋を出る	41
3.2.10	さて、さて、	41
3.2.11	マニュアルなど	41
3.3	Macintosh の telnet で cc2000 を使う	42
3.3.1	まず部屋へ	42
3.3.2	電源を入れる	43
3.3.3	マックの操作	43
3.3.4	端末ソフトの起動	43
3.3.5	さあ、login しよう	44
3.3.6	ちょっとメッセージ	44
3.3.7	パスワードを変える	45
3.3.8	logout をしよう	46
3.3.9	端末ソフトを終了する	47
3.3.10	電源を切る	47
3.3.11	困った時の対処	47
3.3.12	cc2000 以外のホストコンピュータを使う時の注意	48

3.3.13	マニュアルなど	49
3.4	WindowsNT の telnet から cc2000 を使う	50
3.4.1	まず部屋へ	50
3.4.2	電源を入れる	51
3.4.3	WindowsNT にログオンしましょう	52
3.4.4	まずは WindowsNT のパスワードを変更しましょう	52
3.4.5	端末ソフトを起動しましょう	54
3.4.6	ホストコンピュータへの接続	55
3.4.7	さあ、login しよう	56
3.4.8	ちょっとメッセージ	56
3.4.9	パスワードを変える	57
3.4.10	logout をしよう	58
3.4.11	端末ソフトを終了する	58
3.4.12	ログオフ (シャットダウン) しましょう	59
3.4.13	さて、さて、	60
3.4.14	困った時の対処	60
3.4.15	マニュアルなど	61
3.5	Turbo Linux j3.0 を使う	62
3.5.1	まず部屋へ	62
3.5.2	電源を入れる	62
3.5.3	login する	64
3.5.4	ちょっとメッセージ	65
3.5.5	ターミナルはあるかな？	65
3.5.6	パスワードを変える	67
3.5.7	logout する (セッションを終了する)	68
3.5.8	シャットダウンする	69
3.5.9	部屋を出る	70
3.5.10	さて、さて、	70
<b>第 4 章</b>	<b>UNIX それから</b>	<b>71</b>
4.1	基礎知識をもう一度	71
4.1.1	login	71
4.1.2	キー表記	71
4.1.3	カーソル	72
4.2	コマンド	73
4.2.1	コマンドって何だ？	73
4.2.2	プロンプト	73
4.2.3	簡単なコマンド	73
4.2.4	引数とオプションのあるコマンド	74
4.2.5	対話的なコマンドとそのサブコマンド	75
4.2.6	コマンドの使い方を調べる	76
4.2.7	UNIX によるコマンドの違い	78
4.2.8	トラブルからの脱出	78
4.3	シェル	80

4.3.1	コマンド入力時の編集	80
4.3.2	履歴	81
4.3.3	イベント	82
4.4	ファイル	83
4.4.1	でもやっぱりファイルって何？	83
4.4.2	ファイルの一覧を見る	83
4.4.3	試しにファイルを作ってみましょう	84
4.4.4	ファイルの内容を見る	84
4.4.5	ファイル名を変える	85
4.4.6	ファイルの複写	86
4.4.7	ファイルの消去	86
4.5	ファイルを編集する	87
4.5.1	Mule での作業の流れ	87
4.5.2	Mule の起動	87
4.5.3	A. の場合：X ウィンドウ環境での Mule の起動とエラー対策	88
4.5.4	B. の場合：非 X ウィンドウ環境での Mule の起動とエラー対策	89
4.5.5	ファイル名の指定	90
4.5.6	編集	90
4.5.7	ファイルへの保存	92
4.5.8	Mule の終了	92
4.5.9	Mule もっともっと	92
4.6	印刷	95
4.6.1	どんなプリンタがあるか	95
4.6.2	ファイルの印刷	95
4.6.3	印刷状況をチェックする	96
4.6.4	印刷の取消し	97
4.6.5	利用上の注意	97
4.7	状況の変化	98
4.8	ファイルの階層構造	99
4.8.1	ディレクトリ	99
4.8.2	ツリー構造におけるファイル名の表記	102
4.8.3	ディレクトリの扱い	103
4.8.4	ディレクトリを意識したコマンドの書き方	104
<b>第 5 章</b>	<b>ネットワークの世界へようこそ</b>	<b>109</b>
5.1	ネットワークサービス紹介	109
5.1.1	ネットワークサービスと向き合う	109
5.1.2	電子メールって何？	110
5.1.3	ネットニュースって何？	110
5.1.4	WWWって何？	110
5.1.5	京都産業大学のネットワーク	111
5.1.6	インターネットとは？	111
5.1.7	Internet mail サービスってどんなもの？	112
5.1.8	Internet news サービスってどんなもの？	113

5.1.9	WWW サービスってどんなもの？	113
5.1.10	ネットワークでの暮らし方	114
5.2	電子メール準備体操	114
5.2.1	Internet mail アドレスについて	114
5.2.2	計算機センター運用のコンピュータのメールアドレス	115
5.2.3	相手のメールアドレス	116
5.2.4	自分のメールアドレス	116
5.2.5	さあ、本番！	117
5.3	MHE : Mule による電子メールの読み書き	118
5.3.1	はじめに	118
5.3.2	メールを読む	119
5.3.3	メールを書く	122
5.3.4	来たメールの返事を書く	123
5.3.5	メールの整理	125
5.3.6	メールの実体はどこに？	126
5.3.7	メールが来ているかどうか確認する	127
5.3.8	メールの印刷	127
5.3.9	トラブルからの脱出	127
5.3.10	MHE もっともっと	128
5.3.11	まとめ	128
5.3.12	signature	128
5.3.13	メールを書くときの注意	129
5.4	GNUS : Mule によるネットニュースの読み書き	132
5.4.1	用語説明	132
5.4.2	はじめに	132
5.4.3	GNUS の起動	132
5.4.4	記事を読む	133
5.4.5	GNUS を終了する、再起動する	135
5.4.6	ニュースグループを選ぶ	135
5.4.7	記事を投稿する(けどちょっと待てよ)	136
5.4.8	新しい記事を投稿する	137
5.4.9	記事にフォローする	139
5.4.10	記事のキャンセル	140
5.4.11	メールで返事をする	140
5.4.12	古い記事を読み返す	141
5.4.13	記事の保存	142
5.4.14	ちょっとしたテクニック	142
5.4.15	signature	143
5.4.16	記事を書くときの注意	143
5.5	ファイルをコンピュータ間で移動	146
5.5.1	コンピュータウイルス	146
5.5.2	フロッピーディスクの種類	146
5.5.3	ファイルの種類	147
5.5.4	ファイルをフロッピーディスクにコピー	150

5.5.5	UNIX コンピュータ間でファイルのコピー	153
5.5.6	anonymous FTP	155
5.5.7	archie	157
5.6	WWW	158
5.6.1	ホームページ	158
5.6.2	URL	159
5.6.3	URL の検索	159
5.6.4	ブラウザ	160
5.6.5	ブラウザの終了	160
5.6.6	ブラウザの使い方	161
5.6.7	調べたいキーワードがある場合のブラウザの使用例	162
5.6.8	URL を知っている時のブラウザの使用例	164
5.6.9	情報送信	165
5.6.10	トラブルを出さないために	166
5.6.11	あなたも作れるホームページ	167
5.6.12	FTP 再び	167
<b>第 6 章</b>	<b>UNIX もっともっと</b>	<b>169</b>
6.1	ファイルの整理	169
6.1.1	ファイルの大きさを調べる	169
6.1.2	身に覚えのないファイル	171
6.1.3	ファイルを削除する	172
6.1.4	ディスクの利用量をチェックする	172
6.2	情報検索	173
6.2.1	学内の利用者情報の検索	173
6.2.2	学外の組織などの情報を調べる	174
6.3	ファイルのアクセス権	177
6.3.1	アクセス権	177
6.3.2	UNIX におけるアクセス権	177
6.3.3	アクセス権限を調べる	178
6.3.4	アクセス権限を設定する	179
6.4	シェルよもう一度	181
6.4.1	ファイル名の補完	181
6.4.2	対話的なファイル名の補完	182
6.4.3	シェル変数と環境変数	183
6.4.4	リダイレクション	184
6.4.5	パイプ	186
6.4.6	シェルの鬼へのヒント	187
6.4.7	シェルよ永遠に	189
<b>第 7 章</b>	<b>Mule</b>	<b>190</b>
7.1	なにそれ?	190
7.2	Mule の仕組み	190
7.3	これだけ覚えたら Mule は使える!	191
7.4	もうちょっと覚えよう	192

7.5	日本語入力	194
7.5.1	かな漢字変換	194
7.5.2	ためしに変換	195
7.5.3	特殊な文字列の入力	197
7.5.4	まとめ	198
7.5.5	ローマ字入力のヒント	199
7.6	ちょっと中級者	200
7.6.1	単語登録	200
7.6.2	検索	201
7.6.3	画面分割	201
7.6.4	バッファリスト	201
7.6.5	モードライン	202
7.6.6	Dired モード	202
7.7	上級者に向けて	203
7.8	X 環境での Mule	203
7.9	とりあえずここまで	204
<b>第 8 章</b>	<b>はじめての HTML: WWW ページを作ろう</b>	<b>205</b>
8.1	はじめに	205
8.2	HTML を書く前に：忘れちゃいけないこと	205
8.2.1	WWW ページに載せても良いもの	205
8.2.2	WWW ページに載せてはいけないもの	206
8.3	HTML を書くための準備	206
8.3.1	WWW ページ用のディレクトリはどこ？	206
8.3.2	Mule は使える？	207
8.3.3	Netscape などのブラウザは使える？	207
8.3.4	忘れずに！ Mule で Save した後にブラウザで Reload	208
8.4	これだけで WWW ページは書ける：基本編	208
8.4.1	ファイル名の付け方	208
8.4.2	「タグ」って何？	209
8.4.3	重要なタグ	209
8.4.4	特殊文字：そのままでは出ない文字	211
8.4.5	コメント (書くけれども表示させない)	212
8.4.6	改行する	212
8.4.7	段落を区切る	212
8.4.8	区切り線を入れる	213
8.4.9	見出しを付ける	214
8.4.10	文字を修飾する	214
8.4.11	リスト (箇条書き)	215
8.4.12	字下げ (インデント) する	219
8.4.13	書いたものをそのまま表示	219
8.4.14	画像を入れよう (インライン画像)	220
8.4.15	リンクを張ろう	222
8.5	HTML もっともっと	225

8.5.1	他の人はうまく見える? . . . . .	225
8.5.2	背景、本文などの色を指定する . . . . .	225
8.5.3	文字の色を変える . . . . .	226
8.5.4	見出しの位置を変える . . . . .	226
8.5.5	段落の位置を変える . . . . .	227
8.5.6	特定の文字の大きさを変える . . . . .	227
8.5.7	真ん中寄せ、右寄せにする . . . . .	228
8.5.8	簡単な表を作る . . . . .	228
8.6	WWW ページ作成に役立つもの . . . . .	230
8.6.1	HTML を書くお助けツール: html-helper-mode . . . . .	230
8.6.2	HTML 文法チェッカ: jweblint . . . . .	231
8.7	HTML のこれから . . . . .	233
8.7.1	最新の規格: HTML4.0 . . . . .	233
8.7.2	スタイルシートについてちょっとだけ . . . . .	233
8.7.3	次世代の HTML: XML . . . . .	235
8.8	おわりに . . . . .	238
<b>第 9 章</b>	<b>L<sup>A</sup>T<sub>E</sub>X</b>	<b>241</b>
9.1	L <sup>A</sup> T <sub>E</sub> X(らてふ)って何? . . . . .	241
9.1.1	L <sup>A</sup> T <sub>E</sub> X とは? . . . . .	241
9.1.2	L <sup>A</sup> T <sub>E</sub> X の特徴 . . . . .	241
9.1.3	L <sup>A</sup> T <sub>E</sub> X の作業の進めかた . . . . .	241
9.2	それでは使ってみましょう . . . . .	241
9.3	L <sup>A</sup> T <sub>E</sub> X におけるルール . . . . .	247
9.3.1	最低限のルール . . . . .	247
9.3.2	ドキュメントスタイルについて . . . . .	247
9.3.3	見出しの種類 . . . . .	248
9.4	いろいろなコマンドと環境 . . . . .	250
9.4.1	特殊文字 . . . . .	250
9.4.2	特殊文字でもそのまま出力する . . . . .	252
9.4.3	文字の空白 . . . . .	252
9.4.4	改行と改ページ . . . . .	253
9.4.5	水平方向と垂直方向の空白 . . . . .	253
9.4.6	引用 . . . . .	255
9.4.7	箇条書き . . . . .	256
9.4.8	右寄せ、中央寄せ、左寄せ . . . . .	260
9.4.9	文字の大きさ . . . . .	260
9.4.10	書体 . . . . .	261
9.5	表題 . . . . .	262
9.5.1	タイトルの作り方 . . . . .	262
9.5.2	概要の作り方 . . . . .	262
9.6	傍注 . . . . .	263
9.7	脚注 . . . . .	263
9.8	相互参照 . . . . .	263

9.9	箱	265
9.9.1	一行に収まる文字列を囲む	265
9.9.2	複数行にわたる文の箱を作る	267
9.9.3	minipage 環境	268
9.10	表	270
9.11	絵	272
9.12	L <sup>A</sup> T <sub>E</sub> X で扱える記号	275
9.12.1	雑記号	275
9.12.2	空白を空ける文字	275
9.12.3	アクセントなど	275
9.12.4	ヨーロッパ系言語特有の記号	276
9.13	数式環境で使える記号	277
9.13.1	雑記号	277
9.13.2	ギリシャ文字	280
9.13.3	関数	281
9.14	エラーの対処	282
9.14.1	エラーの表示	282
9.14.2	ちょっと違うやり方	284
9.14.3	エラーの種類	285
9.15	部分印刷する方法	286
9.16	自分の命令 (マクロ)	288
9.16.1	簡単な命令を作ってみよう	288
9.16.2	引数を持っている命令の作り方	288
9.16.3	マクロの名前の付け方	288
9.17	ファイルを分けて L <sup>A</sup> T <sub>E</sub> X を使う方法	289
9.18	標準以外のスタイルファイル	289
9.18.1	日本語 L <sup>A</sup> T <sub>E</sub> X 定番スタイル集の使い方	289
<b>第 10 章 AUCT<sub>E</sub>X</b>		<b>290</b>
10.1	AUCT <sub>E</sub> X で L <sup>A</sup> T <sub>E</sub> X 生活が変わる	290
10.2	AUCT <sub>E</sub> X の起動	290
10.3	C-c C-e	291
10.4	C-c {	292
10.5	C-c C-c	292
10.6	C-c ‘	293
10.7	その他の機能	294
10.7.1	部分的なコンパイル	294
10.7.2	ドキュメントの分割編集	294
10.7.3	アウトラインマイナーモード	295
10.7.4	L <sup>A</sup> T <sub>E</sub> X マクロの入力	295
10.7.5	複数行のコメントの付け外し	296
10.7.6	書体の指定	296
10.7.7	数式モードの支援	296
10.8	最後に	298

第 11 章 どこからでも telnet	299
11.1 スロースタート	299
11.1.1 どこから telnet?	299
11.1.2 どこへ telnet?	300
11.1.3 これ以降の読み方	300
11.2 誰かが管理してくれているコンピュータから	300
11.2.1 誰かが管理してくれているコンピュータからの接続に対するトラブル対策	301
11.3 あなたのパソコンから	302
11.3.1 準備段階	302
11.3.2 ダイアルアップ IP 接続	304
11.3.3 ダイアルアップ IP 接続について	305
11.4 ダイアルアップ IP 接続の設定と利用法	307
11.4.1 利用上の注意と制限	307
11.5 Windows95 編	308
11.5.1 用意する物	308
11.5.2 インストール	308
11.6 Windows98 編	315
11.6.1 用意する物	315
11.6.2 インストール	315
11.6.3 ネットワークソフトウェアの入手およびインストール	321
11.7 Macintosh 編	328
11.7.1 用意するもの	328
11.7.2 インストール	329
11.7.3 使い方	335
11.8 慣れている人のために	339
11.8.1 ダイアルアップ IP 接続に関する情報	339
11.9 トラブル対策	341
付録 A 章リファレンス	344
A.1 UNIX コマンド	344
A.1.1 ファイル管理に関するコマンド	345
A.1.2 ファイルに関する雑多なコマンド	346
A.1.3 テキスト処理に関するコマンド	347
A.1.4 プリンタに関するコマンド	349
A.1.5 アクセス権、アクセス制御に関するコマンド	349
A.1.6 マニュアルに関するコマンド	350
A.1.7 雑多なコマンド	350
A.1.8 ファイル圧縮などに関するコマンド	352
A.1.9 プロセスに関するコマンド	353
A.1.10 現在使っているコンピュータに関するコマンド	353
A.1.11 利用者に関するコマンド	354
A.1.12 ネットワークサービスに関するコマンド	355
A.1.13 シェル (tcsh) のサブコマンド	356
A.1.14 索引	358

A.2	UNIX でよく使われる記号など	359
A.2.1	シェル変数の一覧	359
A.2.2	環境変数の一覧	359
A.2.3	リダイレクション記号など	360
A.2.4	ファイル指定のワイルドカードなど	360
A.2.5	コマンド履歴を扱う為の表記法	360
A.2.6	正規表現	361
A.3	Mule コマンド	362
A.3.1	絶対覚えておいた方がいいもの	362
A.3.2	必要に応じて覚えるもの	363
A.4	京都産業大学 FAQ(抄)	371
A.4.1	目次	371
A.4.2	はじめに	372
A.4.3	UNIX 編	372
A.4.4	Mac 編	382
A.4.5	Program 編	382
A.4.6	その他	382
	<b>付録 B 章情報処理教室の利用について</b>	<b>383</b>
	<b>付録 C 章著作権法 (抜粋)</b>	<b>387</b>
	<b>付録 D 章参考文献</b>	<b>394</b>
	<b>付録 E 章トラブル解決</b>	<b>398</b>
E.1	画面が正しく表示されない	398
E.2	フロッピーディスクが取り出せない	398
E.3	コンピュータが止まった	399

# 第1章 インターネット利用に関するガイドライン

京都産業大学

## 1.1 インターネットとは

インターネット (Internet) は、世界中の各組織が運用するネットワークをつなぐネットワークです。インターネットでは、電子メールや World Wide Web など様々な通信機能が提供されており、ボーダレスに利用することができます。

インターネットと既存の通信サービスを比較した場合、最も異なっているのは次の点です。電話や手紙と違い、インターネットそのものを統括し管理している組織は存在しません。もともとインターネットはコンピュータを持っている人や組織が、相互にネットワークを繋ぎ、ここまで発展してきました。使用している技術も互いの組織の事情に応じて発展してきたために、開放されたものが多く、銀行のキャッシュディスペンサのオンラインシステムのように、安全性を高めるための閉鎖性はありません。ときには様々な事故や障害により電子メールが届かなかったり、第三者からの意図的に不正な行為によりセキュリティ (安全) が脅かされるといったことが起こり得ます。

また、インターネットでは世界中の人々を相手に情報を発信したり受信したりすることが可能です。異なる文化や伝統を持った人たちを相手にしたコミュニケーションでは、きちんとしたマナーに基づいた言動をとらないと、文化的な摩擦や誤解が生じてしまいます。

それら为了避免するため、インターネットに関する正しいマナーを身につける必要があります。これはネットワーク上でのエチケットという意味で、『ネチケット』と呼ばれます。この章をよく読んでインターネットを楽しくそして大いに利用してください。

なお、本ガイドラインでは、利用資格など細則において計算機センターの運用する cc 環境を想定して説明しています。

### 1.1.1 インターネットで利用できるサービス

#### 電子メール

電子的なメールサービスです。回覧に相当する機能としてメーリングリストがあります。メーリングリストは特定のメンバー間での情報交換ができます。

#### ネットニュース

公開の電子的な掲示板です。ネットニュースへの投稿は世界中に流されます。

#### WWW

World Wide Web が正式名称です。インターネットの統合サービスで、ページという単位で情報提供が可能です。

#### anonymous ftp

公開ファイルの転送機能です。公開された情報の取得ができます。

## telnet

自分がユーザ ID をもつコンピュータを遠隔から利用できます。

## その他のサービス

インターネットでは常に新しいサービスが誕生（開発）したり廃止されたりしています。本学でもそうですが利用できるサービスは常に変化していますので、ある程度技術動向に敏感になっておくことが必要でしょう。

## 1.2 インターネットの利用手続

### 1.2.1 利用資格の取得

本学のインターネット接続サービス利用者は、以下の手続を経てユーザ ID を取得し利用資格を取得できます。

ユーザ ID とは、利用者をネットワークで識別するための名前であり、1つのユーザ ID は、一人の利用者に特定して発行されます。ユーザ ID は、銀行の口座名のようなものであり、利用者はこれを取得することにより、インターネットの利用が可能となります。

利用資格の消滅等については「1.5.4 利用資格の消滅等」を参照してください。

### 1.2.2 本学関係者への発行

#### 1. 本学の教職員

本学に勤務する専任教員，専任事務職員を指します。

申請は、別紙「計算機利用申請書」によって行います。承認されたユーザ ID は利用者が本学に在籍する期間有効です。

#### 2. 本学の学生

本学に在籍する学部学生，大学院生，留学生，科目等履修生，聴講生を指します。

学部学生には、入学時に全員にユーザ ID が交付されます。

学部学生以外のユーザ ID 取得申請は、「計算機利用申請書」によって行います。承認されたユーザ ID は、利用者が本学に在籍する期間有効です。

#### 3. 本学を定年退職する教員

申請により在職中のユーザ ID を継続して利用できます。

申請は、別紙「計算機利用申請書」によって行います。利用は当該年度の末日まで単年度とし、継続申請があればさらに更新できます。

### 1.2.3 学外者への発行

#### 1. 本学の非常勤講師及び学外講師

本学に勤務する非常勤講師及び学外講師を指します。

担当する授業の遂行において、インターネットを利用するためにユーザ ID が必要な場合、申請によ

りユーザ ID の交付を受けることができます。

申請は、別紙「計算機利用申請書」によって行います。承認されたユーザ ID は、利用者が本学に勤務する期間有効です。

## 2. 客員研究員

本学が受け入れを認めた客員研究員を指します。

申請は、別紙「計算機利用申請書」によって行います。承認されたユーザ ID は受け入れ期間有効です。

## 3. その他

その他、計算機センター長が特に必要と認めたもの。

# 1.3 インターネットを使う上で知っておくべきこと

## 1.3.1 ネチケットガイドライン

ネチケット (Netiquette) とは、ネットワーク上でのエチケットのことを指します。インターネットの利用にあたっては必ず適切なエチケットを身につけた上で行動してください。快適なインターネットの利用は、利用者一人ひとりの行動にかかっています。またネチケットは単なる礼儀上のことではありません。あなたの安全を守るためでもあります。ネチケットを守ることはあなた自身を守ることでもあるのです。

インターネットは、居ながらにして世界中から情報を収集することができます。しかし便利な道具、テクノロジーだとばかり考えてしまうのは正しい理解ではありません。インターネットでは互いに異なる考えを持つ人々が情報を交換するわけですから、そこには考えの違いや文化的背景の違いが存在します。インターネットを便利なテクノロジーだと安易に考えたために、相手の意図することを誤解したり、こちらの意図することが理解してもらえなかったりして、互いに不快な思いをすることがあります。

この様なトラブルに巻き込まれないために、インターネットはテクノロジーではなく、文化そのものであることを理解してください。そこには社会性があり、文化交流があるのです。実社会と同じく相手に不快感を与えないように振る舞ってください。

### 1. 安心感を与えるように

ネットワークでは初対面の人との交信が多く発生します。互いに相手の顔が見えないで交流するわけですから、つねに自分を適切に紹介し、相手に安心感を与えるよう心がける必要があります。そのためには、自分をあだ名で呼んだりせず本名（通称）で紹介すべきです。

### 2. よく考えて

インターネット上での情報交換の多くは文字によって行われます。互いの顔が見えないためニュアンスが伝わらなかったり、個人が持つ文化的背景の違いなどから誤解やトラブルが発生しやすいものです。こちらの意図することが間違いなく相手に伝わるかよく考えて発言するようにしましょう。メールなどは送る前に読み返しましょう。

### 3. 無駄をしない

インターネットには、全体を統括する管理組織は存在せず、互いに接続しあうネットワーク組織の相互協力によって成り立っています。接続にかかる維持費用も互いに負担しあうことで成立しています。

例えば学外に電子メールを送れば、本学とは無関係の組織が運用するネットワークを利用させてもら

うこととなります。

もし大量の情報を遠隔地に発信したり受信したりすると、経路途中のネットワーク組織の回線を使っていますから、その組織に迷惑をかけたか、場合によってはその組織を通信不能な状態にしてしまうことも考えられます。したがって、無駄な通信や大量の通信は極力控えることが必要です。

またメールやニュースにおいて、相手の書いた文章を引用する場合、相手の文章の3分の1以下を目安として引用しましょう。全文引用はネットワークの通信量を増やすことになり全体への迷惑ともなりますし、論点が明確でない文章ともなります。メールやニュースの最後に署名（シグネチャ）を付けることができます。必ずしも付ける必要はないのですが、付けるとすればあなたのオリジナリティのある署名を付けたらよいでしょう。ネットワークの通信量を増やさないためにも、目安として4行以内に記述すべきです。

### 1.3.2 セキュリティ

近年、新聞、TVなどのマスメディアでもよく報道されるように、他の機関が運営するコンピュータネットワークに不正に侵入したり、他人のユーザIDを不正に使用して混乱に陥れたり、情報を不正に入手したりする人々があります。この様な人に対抗するための最も単純で確実な方法は、あなたのユーザIDに適切なパスワードを設定することです。利用者一人のパスワードが盗まれることで、他の利用者にも影響が出るため、これは集合住宅の1世帯の戸締まりをしっかりとするようなものです。

自分は、他人に見られたり取られたりして困るようなものをコンピュータに置いていないから、パスワードを盗まれても被害はないという人がいます。しかしこれは大きな間違いです。ネットワーク犯罪者は、目標とするコンピュータに侵入する前に、いくつかのコンピュータを踏み台として不正行為を行います。あなたのユーザIDは、犯罪のための踏み台となっているかもしれないのです。1996年、米国では踏み台となったコンピュータの所有者にも有罪判決が下されたほどです。

”ユーザIDとパスワードの管理は利用者の義務です。”

これらの管理義務を怠ることにより発生する損害には、あなたが責任を取らなければなりません。

この節には、適切なパスワードを設定するために注意しなければならない事項を掲げています。よく読んで、自分の安全をしっかりと守る習慣をつけてください。

また、利用に際し自分のユーザIDが知らないうちに他人に使われているのではないかといった形跡がある場合、システム管理者（計算機センターの担当者）に連絡をしてください。

### 1.3.3 適切なパスワード

1. 自分だけが知っているオリジナルの文字列
2. 大文字と小文字の両方を含んでいる
3. 文字の他に数字を含んでいる
4. 覚えやすく書き留める必要がない
5. 長さが7～8文字で、誰かに覗かれても覚えられないように、素早くタイプできる

### 1.3.4 使ってはいけないパスワード

1. 空のパスワード（パスワードを設定しない）

2. 6文字以下の短すぎるもの
3. ユーザIDと同じもの
4. 辞書，辞典に載っている言葉
5. 地名をはじめとする固有名詞
6. 人名（自分の名前，家族，友人，知人，よくある人名）
7. ペットの名前
8. ゲームに出てくるような呪文やキャラクター名
9. 映画やテレビドラマの登場人物の名前や呼び名
10. 自分や家族，知人の誕生日
11. コンピュータの名称やコンピュータに関する名称（OSの名前やコマンド名）
12. 自動車のプレートナンバー
13. 健康保険証番号の一部
14. 容易に入手できる自分に関する情報
15. キーボードの文字の並びをそのまま使ったもの
16. 文字や文字列を繰り返したもの

## 1.4 インターネットに関わるには

初めてインターネット使おうとすることはだれしも勇気が必要なものです。本学では，在学中にそのような経験を十分つむことができるよう，教職員が一体となって環境を整備しています。また，先輩たちも初心者の方を温かく迎え入れてくれるでしょう。

どうか物怖じせず勇気を持ってインターネットに参加してください。

### 1.4.1 わからないことがあれば

- (1) インターネット利用に関して，様々にわからないことが出てくるでしょう。まずこの文章をしっかりと読んで理解してください。そして大学が発行している「インターネット利用ガイド」やWWWの情報ページ等のドキュメントに目を通してください。
- (2) それでも分からないことがあれば，学内のニュースグループ `sandai.question` 等に尋ねてみるのが良いでしょう。`sandai.comp` には過去に問い合わせされた膨大な情報が貯えられています。これらに目を通すだけで疑問が解けることもあるでしょう。  
ニュースグループは不特定多数の人が利用しています。ニュースグループに問い合わせることの最大の利点は，疑問や解決方法があなただけのものではなく，利用している全ての人で共有できることです。互いに知識を共有できることはインターネットの最大の利点といえるでしょう。

- (3) 学内のニュースグループに問い合わせれば、大半の疑問は解決できるはずですが、それでも分からなければ、計算機センターの運用補助員にメールを送って尋ねてください。運用補助員の電子メールアドレスは `mics-q@cc.kyoto-su.ac.jp` です。

また、学内の情報処理教室でコンピュータの操作中にトラブルが発生した場合は、運用補助員（黄色い腕章を付けています）に直接尋ねてください。運用補助員が情報処理教室に不在の場合は、情報処理教室にはコードレス電話が備え付けてありますので、計算機センターの受け付けカウンターに連絡してください。受け付けカウンターの内線電話番号は、コードレス電話に貼り付けられています。

## 1.4.2 電子メールやネットニュースの利用

### 1. 電子メールの練習

電子メールを練習するには、最初に自分自身にメールを送り、読む練習をするのが良いでしょう。少し自信が付けば友人相手に互いに練習をしてみましょう。

### 2. ネットニュースの練習

京都産業大学には `sandai.*` という学内のローカルニュースグループがあります。ここで投稿の練習をしたり、疑問に思っていることなどいろいろ尋ねてみるのが良いでしょう。

電子メールやネットニュースで、(^\_^) や (x\_x) や (^; などのフェイスマークをうまく使うのもよいのですが、相手にニュアンスが伝わらないこともあるので過信しないようにしてください。

## メールに関する注意

### 1. チェーンメール

不幸の手紙を知っていると思いますが、これを電子メールに応用したものです。

例えば「 標題 (サブジェクト) が となっているメールは新手のウィルスが入っていますから決して中を見てはいけません。このことをあなたの知り合いに伝えてあげてください」といった内容のメールです。人から人へ広がってゆくことからチェーンメールと呼ばれますが、このようなメールを受け取った場合は、他の人に送ってはいけません。チェーンメールはネットワークの通信を無駄に浪費するだけです。あなたを最後にチェーンメールの伝播を止めてください。

### 2. SPAM メール

電子メールを使い始めると、突然知らない会社などから宣伝のメールが来るのを経験しますが、受取手の迷惑を顧みないダイレクトメールを SPAM メールといいます。

SPAM (スパム) は米国の缶詰め食品会社が売り出した商品でしたが、「いき過ぎたダイレクトメール広告」から、多すぎて嫌なものという評価になり、相手の迷惑を顧みないダイレクトメールに使われ始めた呼称です。

現実には SPAM メールを拒絶する方法はありません。詳しくは、本学 Web ページに解説がありますから、 <http://www.kyoto-su.ac.jp/local/center/announce/index-j.html> を見てください。

### 3. 48 時間ルール

メールを受け取って返事を書く必要がある場合、48 時間を目処に返信しましょう、という暗黙のルールですから、これになるべく沿うようにしてください。逆にメールの返事が 48 時間以内に來ない場合、先方はメールを読んでいないか、読めるような環境にいないかも知れませんが、別の連絡手段を採った方がいいという目安にもなります。

### 1.4.3 WWW の利用

World Wide Web は、ページという単位で情報を受発信する技術です。世界中には、それぞれの目的に沿ったページが作られ公開されています。WWW をもってインターネットだ、と誤解している人もいるくらいポピュラーな存在です。

本学は教育研究機関ですから、ページの内容が研究分野や学習課題を中心としたものであることが必要です。しかし例外的にページ作成者の興味のある分野や趣味といったものに関して情報を発信することを、インターネットの理解促進と情報に関して考える機会を与えるために許しています。

ページ作成者は、作るページが世界中から見られていることを、忘れないでください。意味のある情報とは何か、他人の役に立つ情報は何か、を考えながらあなたの独創的なページを作るように心がけてください。

#### 好ましいページ

1. 研究課題や成果に関するページ
2. 専門外であっても特定分野について探求した成果を公開するページ
3. 独創的な作品、成果に関するページ

#### 好ましくないページ

1. 他人の作ったページにリンクしているだけのページ（独創性がない）
2. 独善的な自己紹介だけのページ（あなたが誰であるかは誰も興味がない）
3. 友人の紹介といってそのページにリンクだけのページ（内容がない）
4. 教育研究機関であることを超えて、過度に趣味に偏ったページ

#### ページ作成に関する細則及び注意

1. 一般公開を前提として

作成するページは一般公開してください。会員制のページやパスワードによる一般公開ではないページ運用は認められません。

これは内容が隠蔽されないようにする措置です。大学運営のために学外に対し公開されない情報は、この限りではありません。また、学術的目的から教員の指導のもとに行われる場合もこの限りではありません。ただし、その場合はその旨をページの中に明記する必要があります。

2. 著作権や商標権など知的財産権の尊重

WWW は文字だけでなく、グラフィックや音声・画像・動画を含むページが多く見られ興味を引きます。それらが無料で利用できるからといって、著作権や特許権など知的財産権が存在しないかのよう思い込む人がいます。インターネット上の情報も、現実社会と同じく各法令で守られています。それらを拝借してあなたのページに張りつけることは著作権等の侵害です。また、雑誌や新聞から画像、マンガ、文章などの一節を転用することも著作権に違反する行為です。

一方、あなたの書いた文章、絵、詩など、WWW ページで公開してもそれらは著作権で守られています。著作権は、同時にあなたの権利を守ることにものなることを忘れないでください。

なお、このガイドラインの最後に「1.7 著作権など知的財産権について」の解説があります。

## 1.5 利用に関わる諸規則

利用者は以下に定める事項を遵守しなければなりません。

### 1.5.1 遵守すべき事項

1. 利用者は利用に際しユーザ ID と、自身で設定したパスワードについて責任を持って管理する義務を負います。
2. ユーザ ID 又はあなたの管理する情報が第三者によって使用・利用あるいは改ざんされた形跡がある場合は、直ちにシステム管理者にその旨を連絡してください。
3. 禁止行為に違反した利用者を発見した場合は、直ちにシステム管理者に連絡しなければなりません。
4. 本学利用者としてふさわしい節度ある態度で利用を心がけてください。
5. システム管理上あるいは運営上の安全を確保するために、定められた利用方法などを守ってください。

### 1.5.2 禁止される行為

次の各行為は禁止されます。

1. 法令に違反する行為又は違反する恐れのある行為
2. 公序良俗に反する行為
3. アドレスを改ざんする行為、ユーザ ID の貸与する行為およびパスワードの管理を怠る行為
4. 著作権、商標権等の知的財産権を侵害する行為
5. 第三者に対する迷惑や不利益を与える行為及び誹謗、中傷など人権を侵害する行為
6. 政治活動、宗教活動及び営利活動
7. その他、インターネット及び他のネットワークの正常な維持、運営を妨げる行為

### 1.5.3 規則違反に対する措置及び罰則

1. 禁止行為を侵した利用者については本人に通知することなく作成文書の削除、禁止行為の停止、ユーザ ID の停止又は取り消しを行うことがあります。
2. 利用者は自身のユーザ ID に関し、パスワードの管理不十分又は第三者の不正使用に起因する全ての損害について責任を負うものとします。
3. 安全基準を充たしていないパスワードを使う利用者については、利用者に通知することなくユーザ ID を停止することがあります。
4. 利用者が前項の「1.5.2 禁止される行為」により故意に本学のサービスを運用停止もしくはそれに近い状態に至らせた場合、本学がこれにより被る損害に相当する賠償を請求することがあります。

## 1.5.4 利用資格の消滅等

### 1. 以下の場合に利用資格は失効します

- 定められた利用期間が満了したとき
- 利用者によって失効申請がなされたとき

失効の場合は、当該ユーザ ID はシステムより削除されます。

### 2. 以下の場合に利用資格は停止されます

- 本学が定める規則に違反したとき
- システム管理上の理由により止むを得ず停止させざるをえない場合

停止処分の場合、その理由となる要因がなくなった時点で、利用資格を回復することが可能です。

### 3. 以下の場合に利用資格は取り消されます

- 本学が定める規則に違反したとき

取り消しの場合、当該ユーザ ID はシステムより削除されます。

## 1.6 本学の責任免責および経費の負担

### 1.6.1 免責

本学におけるネットワークの利用は基本的に自己の責任に基づくものとし、以下の事項について大学は責任を負いません。

1. 天災、テロなどの不慮の事故、システム管理上の予測できない事故によるネットワークの停止、データの損失について、大学はいかなる責任も負わないものとします。
2. 本学のインターネット接続サービス、利用者が提供するサービス、又はそれらを通じて他のネットワークサービスを利用することにより発生した一切の損害について、大学はいかなる責任も負わないものとします。

### 1.6.2 経費の負担

#### 学内施設からの利用

学内施設からのインターネットの利用について、課金制度は設けておりません。

#### 学外からの利用

自宅などから ISP(インターネットサービスプロバイダ：インターネット接続を業務とするネットワーク会社) と契約し、本学の提供する各種サービスを利用しようとする場合、自宅から本学までのネットワーク経路の経費については利用者が負担してください。

また、本学のダイヤルアップサービスに電話回線で接続する場合も、電話料などを利用者が負担してください。

## 本学のダイヤルアップ接続サービスについて

本学では、ダイヤルアップ接続サービスの利用には、現時点で課金制度は設けておりませんが、大学の回線数に限りがあるために、時間帯によっては接続できない場合もあります。自宅などから長時間インターネットを利用しようとする方は、独自にISPと契約することをお勧めします。

また、大学のダイヤルアップ接続回線に長時間接続し設備を独占することが無いよう、できるだけ短時間で利用を終了し、他の利用者に回線を譲るよう心がけてください。利用者一人ひとりの心がけで、限りある資源をよりうまく共有し、楽しく利用しましょう。

## 1.7 著作権などの知的財産権について

(参考文献 内田春康=横山経通・インターネット法 (商事法務研究会・1997年))

他人が持っているバイクを勝手に使ったり、他人の土地を勝手に占拠して小屋を建てたりすることが許されないのは誰にも分かります。そのバイクや土地には、それを持っている人に「所有権」という権利があり、そのバイクや土地を勝手に使用する行為は、所有権を侵害することになるからです。実は、バイクや土地のように形のある物のほかにも、無形の物 - - たとえば、文章、絵画、音楽、写真、マーク、アイデアなど - - - も権利の対象となるのです。この権利が、著作権とか特許権とか商標権といわれるもので、「知的財産権」と総称されます。

このうち「著作権」を例に挙げてお話をしましょう。たとえば、音楽の入ったCDを買って、それをカセットテープに録音して、勝手に販売することはそのCDを作った様々の人たちの権利を侵害することになります。つまり、詞や曲を作った人、それを演奏した人、それをCDにして販売した人たちです。そのCDは、それらの人々が創意工夫し、又資産を投じて作り出されたものであり、その意味での財産的価値を持っています。その財産的価値を保護する権利が著作権という権利なのです。同じことは、雑誌をコピーしてそれを販売するような場合にもいえます。このあたりのことは、皆さんも常識的に知っているでしょう。

ところで、我々がインターネットを利用するようになると、簡単に情報を発信できるようになります。ホームページを開設したり、電子メールを利用することによってです。そこで、ホームページを開設するときにも、上に述べた著作権などの知的財産権に注意をして欲しいのです。もちろん、本学のインターネットは、商行為や営利行為のために使用することが禁じられていますので、ネット上で他人の作った著作物を販売する行為はできないのはいうまでもありません。しかし、営利目的でなくても他人の著作物を無断でホームページ上に掲載することは原則としてできないのです(サーバーに転送可能な形で配置することも著作権の内容になっています。これを「送信可能化権」といいます)。他人の著作物の複製は、自分一人で楽しむなど私的利用に限っては許されるのですが、ホームページへの掲載は、不特定多数が見ることになるので、もはやこの私的利用には当たらないからです。電子メールで特定の個人間で他人の著作物の複製を転送するのは、許されますが、メーリングリストなどで多数の人に送る行為は、許容の範囲ではないといえるでしょう。

次に、もう少し具体的に見ますが、基本的には、自分の創造したもの以外は、他人に権利があり、それらを尊重し、その利用には慎重を要するということは、忘れないで下さい。

### 1. 著作権フリーのホームページ作成用素材(イラスト、写真など)集

著作権がないわけではなく、ホームページ作成用の利用が一般的に許されているにとどまるのです。その範囲を超える利用は著作権を侵害します。

### 2. 写真

他人に著作権がある写真を利用するには、著作権者の承諾が必要です。しかし、証明書用の肖像写真のような創作性のないものには著作物性は認められないので著作権者の承諾は不要です。ただし、肖

像権やパブリシティ権（有名人の氏名肖像による顧客吸引力のもつ経済的利益及び価値を排他的に支配する権利）の問題は別ですから注意が必要です。

### 3. 音楽

社団法人日本音楽著作権協会（JASRAC）の許諾を得、利用料を払えば音楽を利用することができます。

### 4. 新聞雑誌の記事

そのまま引用する場合には、著作権者の承諾が必要です。しかし、内容たる情報そのものには著作権はないので、まったく別の表現による場合は、著作権の侵害には当たりません。

### 5. 利用が許されているもの

新聞雑誌に掲載して発行された政治・経済・社会の時事問題に関する論説（学術的なものは除く）は、承諾を得ないで利用することができます。ただし、利用を禁止する旨の表示がある場合には、承諾を得ないで利用することができません（著作権法 39 条）。

屋外に原作品が恒常的に設置されている美術の著作物は平面的に複製することができます（同 46 条）。ただし、著作者人格権（公表権（18 条 1 項）、氏名表示権（19 条 1 項）、同一性保持権（20 条 1 項））を侵害してはいけません（同 50 条）。

### 6. 引用

公表された著作物は、公正な慣行に従った引用により利用することができますが、報道、研究、批評等の目的上正当な範囲内で行われなければなりません（同 32 条 1 項）。引用に際しては、著作物の出所、著作者名を表示しなければなりません（同 48 条）。

### 7. 氏名権・肖像権

他人の氏名権、肖像権を侵害しないように注意しましょう。他人の名前を偽って称したり、勝手に広告などに使用するのは氏名権の侵害になります。使用する場合は、承諾を受けるのが無難です。

著作物でない他人の所有物の映像・写真（例：他人の持っている自動車の写真）を利用することは、法的には問題がないのが基本ですが、慎重を要します（被写体が高価又は珍しいものである場合など）。

### 8. 商標権

ホームページ上に、商品またはサービスに関する広告等のために他人の商標（製品名称、シンボルマーク、有名人の氏名等で登録されたもの）を無断で使用することはできません。その他、商標の使用には特に慎重を要します。

### 9. リンク

基本的にリンクを張る行為は、自由にできると考えて結構です。ただし、自分のホームページの一部であるかのような張り方や著作権を侵害しているホームページにリンクを張って侵害を助長することは著作権などを侵害する場合があります。

## 1.8 情報処理教室の利用について

計算機を利用しての授業や自習のため、学内に情報処理教室を設置しています。情報処理教室は、次の3つに分類されます。

1. 10号館を中核とした全学共通の情報教育に使用される教室
2. 各号館に設置された10号館のサテライトとして主に学生の自習用に使用される教室
3. 各号館に設置された学部・学科の専門教育に高度に特化し学部・学科で運用管理される教室

これらの情報処理教室に設置している機器は、UNIX サーバの端末として利用できる他、パーソナル・コンピュータ（パソコン）としても利用できます。利用できるソフトウェアは各情報処理教室ごとに異なりますので、その教室を管理する所管の事務室で確認してください。

### 1. 情報処理教室の一覧

建物	教室名	OS 台数（予備）	所管
<b>10号館情報処理教室</b>			
10号館 2階	10201 情報処理教室	WindowsNT/Linux 89台 (5)	計算機センター
10号館 2階	10202 情報処理教室	WindowsNT/Linux 104台 (5)	計算機センター
10号館 2階	10203 情報処理教室	WindowsNT/Linux 47台 (2)	計算機センター
10号館 2階	10204 情報処理教室	WindowsNT/Linux 47台 (2)	計算機センター
10号館 2階	10205 情報処理教室	WindowsNT/Linux 47台 (2)	計算機センター
10号館 2階	10206 情報処理教室	Macintosh 47台 (2)	計算機センター
10号館 3階	10301 情報処理教室	WindowsNT/Linux 61台 (3)	計算機センター
10号館 3階	10302 情報処理教室	WindowsNT/Linux 89台 (5)	計算機センター
10号館 3階	10303 情報処理教室	WindowsNT/Linux 21台	計算機センター
1号館 2階	11 情報処理教室	Macintosh 91台	計算機センター
<b>サテライト教室</b>			
2号館 4階	21 情報処理教室	Solaris 44台	計算機センター
3号館 1階	32 情報処理教室	Macintosh 91台 (6)	計算機センター
5号館 1階	51 情報処理教室	WindowsNT 35台	計算機センター
<b>学部・学科の情報処理教室</b>			
計算機科学研究所棟 2階	C1 情報処理教室	WindowsNT/Linux 20台	計算機科学研究所
計算機科学研究所棟 3階	C2 情報処理教室	Solaris 20台	理学部
計算機科学研究所棟 3階	C3 情報処理教室	WindowsNT/Linux 15台	計算機科学研究所
計算機科学研究所棟 4階	C4 情報処理教室	NEXTSTEP 6台	理学部
3号館 2階	31 情報処理教室	Macintosh6100 44台	外国語学部
5号館 2階	52 情報処理教室	WindowsNT/Linux 39台	経営学部
<b>その他</b>			
図書館	パソコン室	WindowsNT/Linux, Mac 合計 48台	計算機センター

情報処理教室の設備を利用する前には、付録の「情報処理教室の利用について」を一読しておいてください。机、電灯、空調など部屋の設備についての連絡、相談はその教室を管理する所管の事務室にお願いします。コンピュータの操作、動作不良などについての連絡、相談は計算機センターメインカウンターまでお願いします。

## 2. 利用時間

### 1.10 号館情報処理教室

平日 8:45~20:00 土曜日 8:45~17:00

但し、平日 18:00 以降、土曜日 12:00 以降は一部の指定教室のみ利用できます。

### 2. サテライト教室

平日 8:45~18:00 土曜日 8:45~12:00

### 3. 学部・学科の情報処理教室

所管の事務室にお問い合わせください。

### 4. 図書館パソコン室

図書館の開館時間

情報処理教室の利用は授業を優先しますが、授業のない時間帯は研究や自習として自由に利用できます。10号館情報処理教室では自習教室として指示されている教室で利用してください。サテライト教室及び学部・学科の情報処理教室では各建物の掲示板で「情報処理教室利用時間割表」及び補講掲示等で授業有無を確認のうえ、利用してください。なお、日曜日・祝日、夏季一斉休業期間および年末年始期間は使用できません。その他、保守等によるシステム停止日、清掃等による閉室日は、その都度掲示でお知らせします。

## 1.9 利用者サポート

ガイドラインに書かれていたような注意をあなたが遵守して学内のコンピュータを利用していくのであれば、以下のサポートを受けることができます。

### 1.9.1 メインカウンター

計算機センターが管理している情報処理教室のコンピュータを利用するに当たって質問、要望、トラブルがあれば、計算機センター（10号館3階）のメインカウンターの担当員に尋ねてください。

### 1.9.2 困ったときの補助員制度

パソコンを使っていたら急に動かなくなった、印刷しても紙が出てこない等、困ったときは補助員を頼りましょう。

現在<sup>1</sup>、学内のコンピュータ設備に関して2種類<sup>2</sup>の学生アルバイトによる補助員制度があります。利用者からは「コンピュータのことを教えてくれる人」として同じように見えますが、実はそれぞれ役割が違います。

- 計算機運用補助員

情報処理教室を担当しています。消耗品の交換や、トラブル対応が主な仕事です。授業に関する質問にはお答えしかねます。内線 2578 による電話の対応も行っています。愛称を MiCS(みっくす) 補助員といいます。

- コンピュータ基礎実習補助員

---

<sup>1</sup> 1999年3月

<sup>2</sup> 本当は他にもあるのですが、一般的なものではないので省略します。

コンピュータ基礎実習を受講している人向けのアドバイザーです。コンピュータ基礎実習に関する質問を担当しています。それ以外の質問にはお答えしかねます。

まずは部屋の中で補助員の人を探してください。京都産業大学の黄色い腕章をつけていたり、コンピュータの上に補助員と書かれたスタンドを立てていたりしますので、そういう目印を目標に探すといいでしょう。個人の顔を目印に探さないでください。補助員の人を見つけてもその人が担当時間でないかもしれないからです。担当時間でなければその人のプライベートな時間です。補助員として頼るのはご遠慮ください。

部屋に補助員が見当たらない時は情報処理教室でかつ対応時間内であれば備え付けの内線電話から 2578 で連絡が付きます。利用している部屋の名前、コンピュータの番号、状況を伝えて指示を仰いでください。

しかし勤務時間外など電話が使えない時もあります。そんな時でも、コンピュータの電源を切ったりしてはいけません。故障の原因となります。周りの人を頼るか、巻末のトラブル対応を参考に解決を試みてください。

それぞれ補助員の主な勤務は次のようになります。全学休講日や長期休暇中、授業の有無などによって一部例外もあります。ご注意ください。

アルバイトの種類	期間	時間帯	業務内容
計算法運用補助員 (MiCS 補助員)	授業期間平日	9:00 - 18:00	メインカウンターと主な教室には常勤。
	授業期間土曜	9:00 - 13:00	メインカウンターと主な教室には常勤。
	授業期間平日放課後	18:00 - 20:00	メインカウンターに常勤。
	授業期間土曜放課後	13:00 - 17:00	メインカウンターに常勤。
	長期休暇期間	13:00 - 17:00	メインカウンターに常勤。主な教室を巡回。
コンピュータ 基礎実習補助員	-	授業実施時間	11,32 情報処理教室に常勤。

## 第2章 みんなで使うコンピュータ

普通コンピュータというと、ほとんどの人がパーソナルコンピュータ、いわゆるパソコンの事を思い浮かべるでしょう。パソコンは名前が示すとおり、一人で使うように作られているコンピュータです。それに対して「みんなで使うコンピュータ」もあるのです。京都産業大学にもそのようなコンピュータ環境がたくさんあります。ここではそのうち、計算機センターが運用しているマシンで利用できる、二種類についてその概要と利用方法を説明します。

- UNIX<sup>1</sup>
- WindowsNT<sup>2</sup>

UNIX は現在、大学や研究機関で最も多く利用されているコンピュータ環境の一つです。産業用にも非常に多く利用されており、今後も多方面で利用されるでしょう。WindowsNT は企業などで多く利用されているコンピュータ環境の一つです。ワープロや表計算ソフトなどのソフトウェアが充実しています。

### 2.1 計算機センター運用の環境

#### 2.1.1 cc 環境

情報処理教室などのコンピュータやプリンターは計算機センターが運用し京都産業大学の教員、学生に提供しています。そのコンピュータ環境を「cc 環境」と呼ぶことにします。

#### 2.1.2 ユーザ ID とパスワード

一部を除いて情報処理教室のコンピュータは、利用者が利用資格を持っていることを確認します。具体的には「ユーザ ID」と「パスワード」の入力を求めてきます。

ユーザ ID はコンピュータの世界でのあなたの名前で、パスワードはユーザ ID を使っているのが本人であることを証明するものです。ユーザ ID を入力するだけでコンピュータを利用できてしまうと、第三者があなたのユーザ ID を使ってコンピュータを悪用するかもしれません。それでは困るので、ユーザ ID にはそれぞれ、暗証番号である「パスワード」がつけられています。正しいユーザ ID とパスワードの組み合わせを入力して初めてコンピュータが使えます。キャッシュカードと暗証番号のようなものと考えてください。

平成 11 年度以降入学した本学の学生には<sup>3</sup> あらかじめ学生証番号を含んだユーザ ID を登録しています。パスワードも登録されています。ユーザ ID とパスワードはそれぞれ以下のとおりです。受講科目によっては最初の授業で必要としますので携行してください。忘れないように財布の中に入れておくといいかもしれません。

- ユーザ ID     学生証番号の先頭に g をつけたもの。123456 という番号なら g123456。

<sup>1</sup> 「ゆにくくす」と読んでください。

<sup>2</sup> 「ういんどうずえぬていー」と読んでください。

<sup>3</sup> 平成 11 年 3 月まではオリエンテーションを行いユーザ ID を申請、登録していました。

- パスワード 学生証と同時に配布された紙に書かれています。

### 2.1.3 cc 環境の UNIX

cc 環境のうち、UNIX のコンピュータは以下のものです。ホスト名とはネットワーク上の各コンピュータの名前です。以降各コンピュータはホスト名で表現します。

機種名	ホスト名	OS 名	
Ultra Enterprise 3000	cc2000	Solaris	計算機センターマシンルームに設置の Sun 社製コンピュータ。
Ultra1/170E	cssol000 ~ 043	Solaris	2号館 4階 21 情報処理教室に設置の Sun 社製コンピュータ。
Netfinity3000	axt21000 ~ axt46000	Linux	10号館情報処理教室に設置の IBM 社製コンピュータ。

これらの UNIX コンピュータを利用するためには「ユーザ ID」と「パスワード」が必要です。コンピュータはたくさんありますが、これらの UNIX コンピュータを利用するために必要なユーザ ID とパスワードは一つだけです。一つのユーザ ID とパスワードで上記の全てのコンピュータを利用できます。

ユーザ ID とパスワード以外の要素でもこれらの UNIX コンピュータたちは密接に連係しています。これらのコンピュータたちを「cc 環境の UNIX」と呼ぶことにします。

### 2.1.4 cc 環境の WindowsNT

cc 環境のうち、WindowsNT のコンピュータは以下のものです。

機種名	ホスト名	
Netfinity3000	axt21000 ~ axt46000	10号館情報処理教室に設置の IBM 社製 コンピュータ。
DigitalPC200i	cent001 ~ cent035	5号館 1階 51 情報処理教室に設置の DEC 社製 コンピュータ。
VectraVL 6/300	libw001 ~ libw024	図書館パソコン室に設置の HP 社製 コンピュータ。

上記のコンピュータを利用するためにも「ユーザ ID」と「パスワード」必要になります。たくさんの WindowsNT のコンピュータがありますが、これらのコンピュータを使うために必要なユーザ ID とパスワードは一つだけです。このコンピュータたちを「cc 環境の WindowsNT」と呼ぶことにします。

### 2.1.5 UNIX と WindowsNT のユーザ ID

UNIX と WindowsNT を利用する際にユーザ ID とパスワードの入力が必要なことは述べました。話がややこしくなるので今まで伏せていましたが、実は UNIX と WindowsNT は違うパスワード管理になっています。ユーザ ID は同じですが、UNIX のパスワードでは WindowsNT は使えません。その逆も同じです。2つもパスワードを覚えたくない人は UNIX と WindowsNT のパスワードを同じものに揃えるのがいいでしょう。

## 2.1.6 情報処理教室のコンピュータ環境

- UNIX (サーバ機。計算機センターに設置されている大人数での利用を対象にしたもの。ネットワーク経由で利用する。以下サーバ機と記述。)
- UNIX (クライアント機。情報処理教室に設置されている、一人での利用を対象にしたもの。機械の見た目はパソコン。機械の前に座って利用する。以下単に UNIX と記述。)
- Windows NT
- MacOS

情報処理教室で利用できるコンピュータは大きくこれらの4種類に分けられます。情報処理教室に置いてあるコンピュータはUNIX、WindowsNT、MacOSです。それぞれの使い方はここでは触れません。これらのコンピュータの関係を説明します。

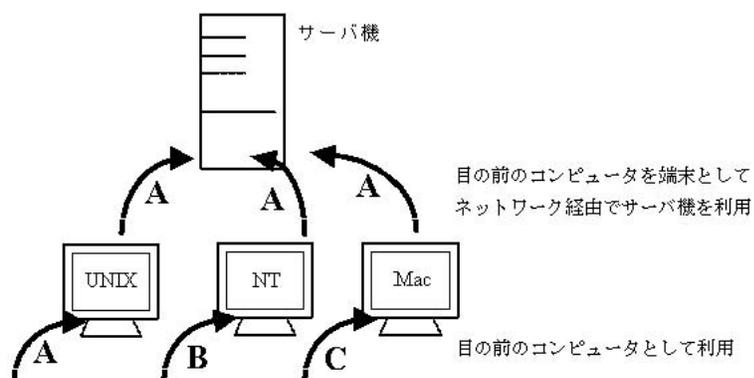
まずは情報処理教室へ行き、コンピュータの電源を入れます。ここで、UNIX と WindowsNT はユーザ ID とパスワードの入力が必要です。MacOS は必要ありません。これで情報処理教室のコンピュータを使うことができます。

さて、更に情報処理教室のコンピュータを端末としてネットワーク越しにサーバ機を利用することができます。その時もやはりユーザ ID とパスワードの入力が必要です。

ユーザ ID を入力して使っている間はその利用者の責任でコンピュータが動いています。コンピュータを使い終わった時は正しい方法で終了の手順を行ってください。いきなり電源を切るとコンピュータが故障しますのでやめてください。また、終了せずに放置すると次の利用者が使えなくて迷惑ですし、あなたの権限でコンピュータが使えますからパスワードを教えたのと同じ事になってしまいます。

先の話とまとめると次の図のようになります。さて、図にはA、B、Cと印が付いています。これは認証の種類を現しています。

- A...UNIX のユーザ ID とパスワード
- B...WindowsNT のユーザ ID とパスワード
- C... 認証なし



例えば WindowsNT のコンピュータからサーバ機に接続するには次のようになります。

1. コンピュータの電源を入れ、WindowsNT を起動する。
2. WindowsNT のユーザ ID とパスワードを用いて WindowsNT の利用開始。

3. WindowsNT の端末ソフトを用いてサーバ機に接続。
4. UNIX のユーザ ID とパスワードを用いてサーバ機の利用開始。
5. サーバ機の利用。
6. サーバ機の終了手続き。
7. WindowsNT の終了手続き。
8. コンピュータの電源を切る（一部のコンピュータは自動的に電源が切れます）。

### 2.1.7 電子メールアドレス

インターネットのサービスの中でも重要な位置を占める電子メール。勿論あなたも利用されることでしょう。友達との連絡や授業の課題提出、三回生になると就職活動でも利用すると思います。さて、電子メールにも通常の手紙と同じように住所が存在します。それが電子メールアドレス（E メールアドレスともいわれます）です。

本学の電子メールアドレスは次のような形をしています。

ユーザ ID@cc.kyoto-su.ac.jp

@は「あっとまーく」と読み、英語の at のような意味です。cc.kyoto-su.ac.jp に所属するユーザ ID の人を表しています。cc.kyoto-su.ac.jp という部分は、日本 (jp) の大学関係 (academy) の京都産業大学 (kyoto sangyo university) の cc というグループという意味です。もし、あなたのユーザ ID が g123456 だったとしたら、電子メールアドレスは g123456@cc.kyoto-su.ac.jp となります。

電子メールアドレスはあなたのインターネット社会での住所であり、名前であることとなります。インターネット社会でのあなたの発言はこの電子メールアドレスの元で行われます。

### 2.1.8 ユーザ ID の変更

さて、ユーザ ID はあなたの電子メールアドレスの一部にもなることがわかりました。では g123456 といった名前ではなく、自分の好きな名前を使いたいと思いませんか？そのほうが愛着も湧くというものです。そこでユーザ ID は一度だけ変更できるようになっています。ただし、あわてて変更せず、次の点をよく読んでください。

- 平成 10 年度までの登録ユーザは既に自分で決めたユーザ ID を使っているので変更できません。
- 変更できるのは一度きりです。再度変更はできません。
- 電子メールアドレスも変わります。電話番号を変更したときと同様、電子メールアドレスを教えた人に連絡が必要です。ですから変更するならなるべく早く、本格的に使い始める前にした方がいいでしょう。
- 就職活動などで電子メールを使うことになるので、あまり突拍子もないユーザ ID をつけることはやめましょう。
- 既に他の人が使っているユーザ ID は使えません。

以下に島田 宏明(しまだ ひろあき)さんを例にとって、これからユーザ ID を決める人達の参考の為に典型的なユーザ ID のパターンを列挙しておきます。

---

名字もしくは名前	shimada , simada , hiroaki
その組合せ	shimadah , hshimada , shiroaki , hiroakis , simahiro
縮めて	shima , sima , hiro , aki
ニックネーム	shimasan , simasan , shima3 , simachan , simachon , shimachi
ミドルネーム (?) を入れて	has

---

さて、変更方法ですが、2通りの方法があります。一つは計算機センターメインカウンターで変更を申し込む方法、もう一つは自分で Web ページから変更する方法です。メインカウンターで申し込む方法は簡単ですが、混雑しているとかかなり待たされますので Web から変更することをお勧めします。

情報処理教室のコンピュータから京都産業大学のホームページを表示します。そして「情報サービス」を選択するといろいろな情報の目次が表示されますので、その中から「ユーザサービス」を選択してください。そこに説明が書かれていますのでそれに従ってください。

### 2.1.9 パスワードの変更

パスワードはいつでも、何度でも変更ができます。同じパスワードを使いつづけるのはセキュリティ上危険ですので、定期的に変更するようにしてください。パスワードを他人に知られることへの防御策になります。なお、計算機センターから与えられたパスワードは2ヶ月以内に変更してください。変更されない場合、利用されていないものとみなします。利用されないユーザ ID とパスワードを放置することはセキュリティ上危険ですのでパスワードはロックされ、使えなくなります。ロックされた場合、計算機センターメインカウンターに行くことで再開できます。

パスワードの変更方法は先に紹介した「ユーザサービス」内のパスワード変更を利用するか、後述の `yppasswd` コマンドを使います。

### 2.1.10 パスワードの安全対策

4 ページ 1.3.2 で説明したように、パスワードは、本人確認のための秘密のキーワードです。毎回、あなたはコンピュータを利用する際にユーザ ID を計算機に通知します。

ところで最近パスワードを調べて悪意に満ちた事をする人がいます。このような人達のために我々が防御策を立てていかななくてはならないのは大変馬鹿馬鹿しい事ですが、それを怠って悪意に満ちた人の攻撃を受けても困ります。そのようなことがないように、まずパスワードを誰にも推測されにくいものにする事を強く推薦します。

#### あなたのユーザ ID の価値

もしかすると、今あなたは「自分のパスワードなんてばれても構わない。勝手に使われて特に支障は無い。自分の名前を使おうとする人は居ない。」などと楽観視していませんか？

- 貴方のユーザ ID で不当な発言をするかもしれません。例えば先生にいたずらメールを送る、インターネット上で人を誹謗中傷する、等。
- 貴方のユーザ ID を踏み台にして別のパスワード入手を試みる人もいます。

- インターネットではクレジットカードを使って買い物ができるサービスもあります。貴方のユーザ ID を使ってネットワークを流れるクレジットカードの番号を盗む人もいます。
- 貴方のユーザ ID とパスワードを人に売る人だっているでしょう。

このようにあなたのユーザ ID には不正に利用する人の隠れ蓑等、十分な価値があるのです。また、ほとんどの人にとって、利用できるユーザ ID が誰のものかは関係ありません。誰のでもいいから使えればいいのです。従って、不正利用しようとする全員があなたのユーザ ID を狙っていると言っても過言ではありません。そして、この不正利用はインターネットに接続されている全てのコンピュータから試みる事が可能です。つまり、ユーザ ID を盗もうとしている人は世界中にいます。不正利用に巻き込まれないよう、パスワードの管理はしっかり行いましょう。もし、万が一不正利用された場合、あなたのユーザ ID で行われた事の責任はパスワードの管理を怠った貴方にかかってきます。

### 悪いパスワード例

不正利用を試みる人は、誰かのユーザ ID に対して「これだ！」と思うパスワードを順に試していきます。そしてたまたま試したパスワードが本物のパスワードと一致するのを期待するわけです。この方法を人の手で実現するのは大変ですが、コンピュータに代行させると十分可能です。例えば英単語ですと、辞書 1 冊分でも数時間で終了します。また、パスワードに利用できる文字を総当りで試す場合もあります。パスワードに使える文字はアルファベット小文字（26 種類）と大文字（26 種類）、数字（10 種類）、記号（32 種類（!"#\$%&'()\*~|'{}+\*<>?\_~^@\[];:,. /））ですが、覚えやすいパスワードにしようと数字しか使わなかったとしましょう。すると、8 文字のパスワードでも 1 億通りの組み合わせにしかありません。1 億という数はコンピュータのスピードにとっては全然問題ない数なので、簡単に総当りでパスワードはばれてしまいます。これが全種類の文字を使うと、8 文字なら 94 種類の 8 乗。つまり 6000 兆通りもの組み合わせになります。これだと総当りしても膨大な時間を要します。

しかし、パスワード破りのテクニックも（困ったことに）年々レベルアップしています。6000 兆通りを試すことは不可能でも、人間心理を考えて確率の高いものを先に試すのです。例えばその人のユーザ ID や本名に含まれている文字の大文字・小文字と数字だけで総当りする、先頭だけ大文字に変えたもの、最後に数桁の数字を加えたもの、などです。

以上のような手口で試されるであろう文字列が悪いパスワードです。例えば、

- ユーザ ID と全く同じ、もしくはユーザ ID に数字をちょっと付け足したくらい。
- 自分や知り合い、有名人の名字、名前、誕生日、電話番号。
- 数字だけ。
- 単語（英語・国語・地名・人名辞典等あらゆる辞書に載っている単語全てです）
- abcdefg,qwerty,asdfgh,zxcvbn,123456 などキーボードの並びを順に打ったもの
- これらを逆順にしたもの（分かり難くなりそうですがコンピュータの前には無力です）

このようなものは簡単に解読されると思ってください。なお、計算機センターでは定期的にパスワードチェックを行います。解読できたパスワードの利用者には警告の電子メールが送られますのでこれを受け取られた方は大至急パスワードを変更してください。一定期間変更されない場合、利用を停止します。この場合、計算機センターメインカウンターで解除できます。

## 良いパスワード例

辞典に載っているような単語の総当たりではとても当たらないものを考えましょう。

- 大文字、小文字、数字、記号を全て混ぜる。
- できるだけ長い、つまり8文字使ったもの。
- 個人の情報から類推できないもの。
- ikA1000(いかやきせんべい)、doBu6da4(どぶろくだよん)、nI+2=yoN(2足す2は4)、83gA:Da(だるまさんがころんだ)等、語呂合わせで考えてもいいかもしれません。
- たまに(数ヶ月に一度くらい)変更する。
- 後ろで見られていてもわからないくらい早く入力できることも重要です。

パスワードは一度判りにくいものに決めてしまった後でも、半年おきなど定期的に変更することが大切です。なぜならパスワードを変えるということは、もしパスワードがばれて誰かに使われていたとしても被害を最小限に押さえる事ができるからです。パスワードを変えてしまえばそれまで不正に使っていた人は使えなくなりますから。

もしも忘れてしまった場合は計算機センターのメインカウンターに学生証を持って来て頂ければ、新しいパスワードでやり直すことができますから安心してください。逆に覚え易くしたい余りに、簡単過ぎるパスワードにしてしまったり、一度決めたきり変更しないで使い続ける方が危険です。注意してください。

これ以降は以下の事項についてそれぞれ説明します。あなたの理解度に応じて必要な所を「つまみ読み」してください。

- login と logout
- キーボード
- ウィンドウ環境
- どこから login?
- コンピュータウィルス

この章の最後には、具体的に UNIX コンピュータを利用する時にどの部屋に行って作業するべきかについて説明します。ここの所は特に重要です。では、3まであなたの理解度に応じて読み進みましょう。少々読むのが面倒でもくじけずに、頑張ってください。

## 2.2 login(ログイン) と logout(ログアウト)

UNIX コンピュータを利用する際には、先に述べたユーザ ID を使って、誰が UNIX コンピュータをこれから利用するのかをその都度確認する作業があります。これを login と呼びます。また、利用が終了したときに今まで使っていたコンピュータを解放して、他の人が使えるようにするために行う作業を logout と呼びます。login してから logout するまでの間を「セッション」などと呼ぶ場合もあります。login する代わりに「セッションを開始する」、logout する代わりに「セッションを終了する」などと表現するときもあるでしょう。

この login を実際に行なうには、様々な方法があります。後でまとめて説明します。

## 2.3 キーボード

コンピュータで作業する為には、キーボードによって文字を打ち込まなければならない事が多いと思います。特に UNIX コンピュータは殆どの場合「あれをしろ」とか「これをしろ」と、決められた文法で指示を（呪文のような）文字列としてコンピュータに入力する事によって操作しますから、キーボードは操作には欠かせません。

- キーボードのキーを、入力したい文字の順番に押して行くことを「タイプ」もしくは「打ち込み」などと言います。コンピュータの操作関係の説明で、「～～とタイプする」もしくは「～～と打ち込む」などと書いてあった場合は、そのとおりキーを押すのだなと解釈してください。
- 単語と単語を分けるために空白を一つ入力したいときは「スペース」キーを押します。キーボード最下段の何も書かれていない横長のキーです。
- 普通にアルファベットのキーをタイプしただけだと一般的には英文字の小文字 (a b c など) が入力されます。英大文字 (A B C など) を入力したい場合はシフトキーを押しながらアルファベットのキーを押すことによって行います。数字キーは図のように「1」の上に「！」が印字されていますが、このようなキーはふつうに押すと「1」が、シフトキーを押しながら押すと「！」が入力されます。シフトキーはキーボード上に二つあり、左側ならアルファベットの並びの最も左下にある「Z」のすぐ左、右側ならアルファベットの並びの最も右下にある「M」のいくつか右にあります。キーには「shift」などと書いてあるでしょう。キーボードによっては上矢印が書いてあったりします。

Esc	!	@	#
	1	2	3
Tab	Q	W	
Ctrl	A	S	
Shift	Z	X	

- 文字を入力していて次の行に行きたくなくなったり、コマンド（後述）を入力し終って実行させたいときには「リターン」キーを押します。改行キーは右側のシフトキーの真上、もしくは二段上くらいにあります。「Return」「改行」などと書いてあります。
- 打ち間違えた文字の修正には delete キーを利用します。一度 delete キーを押すと、一つ前の文字が消えます。delete キーは大抵改行 (return) キーの真上、もしくは二段上くらいにあります。一般的には「DEL」「delete」などと書いてあります。「Back space」の場合もありますね。文字ではなく  や、などと絵で表現してある場合もあります。

- UNIX で時々「コントロールC」などという表現を使うことがあります。表記では Control - C や C-c などと書かれている事が多いようです。この場合はコントロールキーを押しながらアルファベットの C キーを押すことを意味しています。コントロールキーは、大抵アルファベットの A キーの左か、左シフトキーの下かどちらかにあります。キーには大抵「Control」「CTRL」などと書いてあります。

## 2.4 ウィンドウ環境

最近のコンピュータでは当たり前になってきた、ウィンドウ環境の操作方法について説明しておきます。ウィンドウ環境と曖昧に呼んでいますが、厳密な呼び名はありません。最近のコンピュータはユーザの操作がやりやすい様に、画面に絵柄を表示します。つまりそれぞれの絵柄に意味を持たせ、絵で表示されるものを操作させることでコンピュータにユーザの希望を伝えようというわけです。勿論コンピュータがユーザに伝えたい希望（もしくは情報）も絵で表示できるわけで、要はコンピュータとユーザの対話に絵柄を利用しよう、という訳です。このアイディアはアメリカから輸入されたものですから、一般にはグラフィカルユーザインタフェース (GUI) などと英語で呼ばれています。

コンピュータの種類に応じて様々なウィンドウ環境が使われています。UNIX では X Window と呼ばれるシステムをベースにしたウィンドウ環境がほとんどです。パソコンでは Microsoft Windows や Macintosh のいずれかでしょう。

これらのウィンドウ環境でキーボードに代わってコンピュータの操作に活躍するのが次に説明する「マウス」です。

### 2.4.1 マウス

「マウス」とはキーボードの横に置いてある以下のようなものです。



図 2.1 マウス外観

幾つかのボタンが上の方（奥の方）に付いており、その向こうに細い電線が付いています。ボタンの数はコンピュータの種類によってまちまちです。手前側を頭に見立てると、向こうのおしり側に尻尾が付いている形からネズミのつもりでマウスと呼んでいます。利き手で握って机もしくは専用の台の上を前後左右に滑らせて使います。

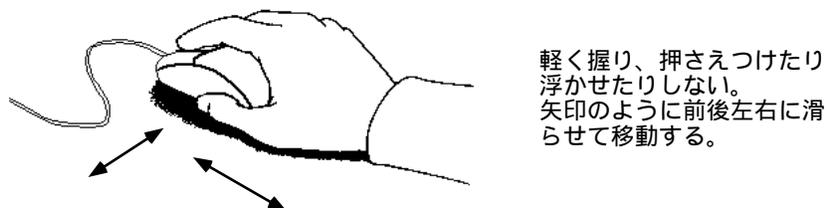


図 2.2 マウスの握り方と移動

初めはちょっと使いにくいと感じるでしょう。これは練習して慣れるしかありません。滑らせて動かすことに注意してください。強く握って押さえついたり、前後左右に動かす時に浮かせたりしないように注意してください。マウスはキーボードでは指示できない、画面に表示されている「もの」に対して指示を

与えるために利用します。画面をよく見ると、マウスの前後左右の動きに連動して画面上を上下左右に動く小さな矢印 （もしくは小さな **X** 印）が見つかるでしょう。これを指示したい対象（絵柄）の上まで運んでいって、マウスボタンの操作によって動作をその対象に対して起こさせるのです。

- 小さな矢印  を「指し示すもの」という意味でポインタ、マウスポインタと呼びます。マウスポインタは状況に応じて形が変わります。例えば **X** などになることもありますが、これも同じくマウスポインタと表現します。
- マウスにはボタンが幾つか付いています。それぞれ左ボタン、中ボタン、右ボタンなどと呼んでいます。
- マウスのボタンを押しっぱなしにすることを「プレス」と呼んでいます。
- マウスのボタンを押してすぐ離すことを「クリック」と呼んでいます。
- マウスのボタンを二度続けて短い時間間隔でクリックする（要するに続けて二回クリックすること）を「ダブルクリック」と呼んでいます。一度目のクリックと二度目のクリックの間にマウスが移動しないように注意してください。ボタンを強く押さえている人は力が余ってマウスも一緒に押ししてしまうようです。
- マウスのボタンをプレスして、そのまま移動することを「ドラッグ」と呼んでいます。目標の場所までマウスポインタが移動したら、ボタンを離します。
- 上記のクリック、ドラッグなどの操作の説明で、特にボタンを指定せずに「クリックする」と表現してある場合は左ボタンで操作を行なうことを意味しています。ボタンが一つしかないマウスでは当然「その」ボタンです。
- マウスを動かして、例えばもっと右端までポインタを移動したいのにマウスを更に右に動かすスペースがなくなってしまう時があるかもしれません。この場合は一旦マウスを持ち上げて少し左にマウスを運び、それからまたマウスを右に滑らせます。
- ポインタを選びたいものの絵柄の上まで持って行って、どのボタンでクリックすると（もしくはダブルクリックすると）どのような反応をするかは状況に応じてまちまちです。一応標準的な規約はあって、このような絵柄のものに対してクリックするとどう反応するという事が決まってはいるのですが、そう完全でもありません。これについては慣れて行くしかないでしょう。

## 2.4.2 ウィンドウ環境の画面

では、Ultra1 の login 直後の標準的な画面表示を例に、ウィンドウ環境のものの呼び方を説明します。

### ウィンドウ

なにはともあれウィンドウ（窓）です。ウィンドウ環境では一般的に次のような四角い枠の中にさまざまな情報が表示されます。

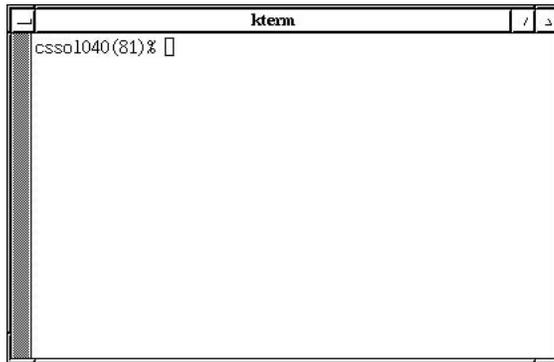


図 2.3 ウィンドウの例

このような枠（窓？）のことを「ウィンドウ」と呼んでいます。ウィンドウはコンピュータの画面の中に一つだけとは限りません。大抵幾つも表示させて使うことになるでしょう。つまり一つの画面の中に複数の小さな画面が幾つも作れる、という訳です。

#### ウィンドウの移動

ウィンドウは画面上のどこにでも配置することができます。ウィンドウの枠の上の部分を見ると、そこにはタイトルが表示されている部分があると思います。



図 2.4 ウィンドウのタイトル部分

このタイトル文字の真上にマウスポインタを移動して、そのままマウスの左ボタンを押してドラッグ（ボタンを押しっぱなしにしたまま移動）します。するとポインタの移動と共にウィンドウ、もしくはウィンドウの外枠が移動するのが判るでしょう。目的の地点までドラッグしたら、マウスのボタンを離します。

#### ウィンドウの上下関係を変える

ウィンドウを同時に二つ三つ出す程度ならせいぜい重ならない場所に移動するだけでいいのですが、画面の大きさには限りがあって、大抵ウィンドウは重なりあってしまいます。下の図のような状態です。

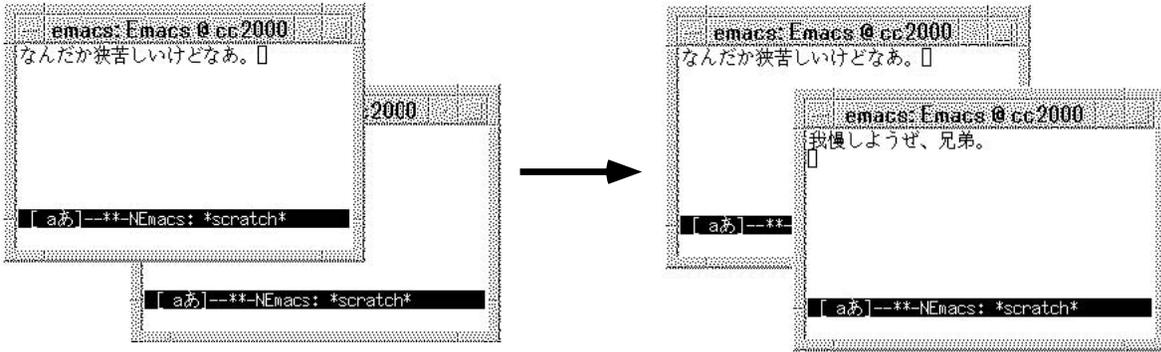


図 2.5 ウィンドウの上下関係を変える

ウィンドウの上下関係を変えたい場合は、自分が上に持ってきていたいと思うウィンドウのどこかの部分をマウスの左ボタンでクリックします。簡単ですね。この図では初め左上のウィンドウが右下のウィンドウを下敷にしていたのですが、右下のウィンドウの見えている部分をクリックすることによってその上下関係が逆転したところを示しています。

#### ウィンドウを選ぶ

現在のウィンドウ環境ではコンピュータに対する指示は、画面上のたった一つのもの（例えばウィンドウ）だけで、複数同時には行なえません。つまりどのような操作にしても、その指示対象はたった一つのものに向けられているのです。そのため、常にウィンドウ環境ではたったひとつのウィンドウが指示対象として「選ばれて」います。この選ばれたウィンドウのことを「アクティブウィンドウ」と呼んでいます。「そのウィンドウを指示対象として選ぶ」ということを「そのウィンドウをアクティブにする」などともいいます。

目標のウィンドウをアクティブにするのは簡単で、単にそのウィンドウを一度マウスの左ボタンでクリックするだけです。アクティブなウィンドウは常に画面上ではもっとも上に来ています。先のウィンドウの上下関係を変える例では、アクティブなウィンドウも右側のものに切り替わっているというわけです。

### 2.4.3 メニュー

ウィンドウ環境で指示を与える方法として、「メニュー」によるものがあります。つまり一覧から自分の望みのものを選ぶというやり方です。勿論一覧の中に自分の望みの指示が含まれていなければいけません。メニューの形はいろいろあって、そのすべてについて網羅的に紹介することはできません。ただ、すべてに共通のやり方としては、項目ないしは絵柄、場合によっては画面の背景などをマウスでクリックすればそこから一覧が湧いて出て、その中から自分の望みの指示をマウスでクリックすることによって選択します。ウィンドウシステムによっては項目ないしは絵柄をプレスすればそこから一覧が湧いて出て、その中から自分の望みの指示の項目までドラッグすることによって選択する場合があります。

ここでは例として Ultra1 に見られるメニューを挙げます。

Root Menu
KTerm
Mule
Netscape
Applications ▾
MO Disk ▾
Floppy Disk ▾
Restart fvwm
Exit

図 2.6 メニューの例：Ultra1 のルートメニュー

このメニューにはたくさんの項目が用意されています。ここで例えば「Exit」の行までマウスポインタを移動させてからマウスボタンを離すと、それはつまり「ルートメニューから Exit を選択し、実行する」ということを意味しています。

#### 2.4.4 ボタン

ウインドウ環境で指示を与える方法として「ボタン」<sup>4</sup> によるものがあります。つまりあるボタンを押すことによって、そこに割り当てられた動作を指示するという訳です。ボタンの形や配置はいろいろあって、そのすべてについて網羅的に紹介することはできません。

ここでは例として xv のデータ保存時に見られるボタンを挙げます。



図 2.7 ボタンの例：xv のメッセージの確認

このウインドウには「OK」と「Cancel」の二つのボタンがあります。ここで利用者はそのどちらを実行するかをマウスの左ボタンでクリックすることによってコンピュータに指示するというわけです。

ちなみにこのウインドウのように、ユーザに強制的に指示を要求する小さなウインドウのことを「ダイアログ」と呼ぶ場合もあります。

#### 2.4.5 ウインドウ環境のトラブル傾向と対策

ここではよく起きるトラブルの傾向とその対策を示します。但しこれですべてのトラブルが解決する訳ではありませんし、本当にコンピュータが故障したのかもしれませんが。ですから対処は慎重に、ゆっくり確かめながら行ってください。

<sup>4</sup> マウスのボタンじゃなく、画面のボタンです！

とにかくトラブルなんですけど！

これ以降をじっくり読んで、自分が該当しているトラブルがあるかどうか調べてください。どうにも当てはまらない場合はまずよく知っていそうな人に聞くのがいいでしょう。それでも解決しない場合は計算機センターのメインカウンターまで連絡してください。気が短い人やパーソナルコンピュータを使い慣れている人の中には突然コンピュータの電源を切ったりする人もいますが、決してそれだけはしないでください。特に UNIX コンピュータは非常に壊れやすく、突然電源を切ったりすると次に電源を入れても二度と立ち上がらなくなることが多々あります。

キーボードをタイプしても、目的の場所に文字が入力されない

目的のウィンドウがアクティブになっていないのではないのでしょうか。目的のウィンドウを一度左ボタンでクリックしてください。

それでも直らない場合は C-q (コントロールキーを押しながら「Q」キーを押す) してみてください。

キーボードをタイプしたらカタカナもしくは変な文字が入力される

キーボード上の「かな」キーを押してしまったのではありませんか？もう一度「かな」キーを押すか「英数」キーを押すなどして解除してください。

目的のウィンドウが他のウィンドウの下敷になって見えなくなってしまった

仕方がありません。上にあるウィンドウを移動させて下敷になっているウィンドウが見えるように場所を変えます。ひょっとするとウィンドウの背景(どのウィンドウの上でもないところ)にマウスポインタを持って行って右ボタンをプレスすればウィンドウの一覧メニューが現れて目的のウィンドウを選択できるかもしれません。それらを使って奥にあるウィンドウを前に出してくることもできます。

## 2.5 どこから login?

UNIX コンピュータを利用するには以下の二つの形態があります。

- UNIX コンピュータの前に直接座ってそれを利用する。
- UNIX コンピュータを、離れた所にあるパソコンから遠隔操作で利用する。

UNIX コンピュータを遠隔操作で利用するという事は、UNIX コンピュータとは離れた所にある別のコンピュータから UNIX コンピュータに login するという事を意味しています。

当然のことですが、UNIX コンピュータを利用するには直接その前に座って利用の方が操作は簡単です。しかし UNIX コンピュータは台数に限りがある上、設置場所が限られていますから、学内のどこに居ても使えるというわけではありません。それに対してパソコンであれば多くの建物に共同利用のものが置いてあります。様々な事情<sup>5</sup> から UNIX コンピュータの前に直接座るのではなく、これらのパソコンの前に座って作業したい場合があるでしょう。そのような時でも遠隔操作によってパソコン経由で UNIX コンピュータを利用することができるのです。

但しこの場合は UNIX コンピュータそのものの使い方に加えて、そのパソコンの使い方もある程度は理解する必要が有りますから少し操作は難しくなるかもしれません。初心者は可能であれば UNIX コンピュータの前に直接座って操作することをお勧めします。

具体的に cc 環境の UNIX コンピュータを利用するには、以下のような方法となるでしょう。自分の事情に合わせてどの部屋から利用するかを決めてください。それによって UNIX コンピュータの前に直接座るか、パソコンの前から遠隔操作するかが決まります。

種別	場所	部屋名	設置されているコンピュータ
直接操作	2号館 4階	21 情報処理教室	Ultra1/170E
	10号館各階	10201 ~ 10303 情報処理教室	Netfinity3000(ただし Linux を使用したとき)
遠隔操作	10号館各階	10201 ~ 10303 情報処理教室	Netfinity3000(ただし WindowsNT を使用したとき)
	1号館 2階	11 情報処理教室	PowerMacintosh 6100/66
	3号館 1階	32 情報処理教室	PowerMacintosh 8500
	5号館 1階	51 情報処理教室	DigitalPC200i
	図書館 1階	パソコン室	PowerMaintosh 6100
	図書館 1階	パソコン室	PowerMaintosh 7200
	図書館 1階	パソコン室	HP Vectra VL 6/300

もしも直接 UNIX コンピュータを利用しようと考えた場合は、具体的には 21 情報処理教室の Ultra1 に向かう事になります。これ以降の説明は読み飛ばして、3 から読み続けてください。

もしも遠隔操作で UNIX コンピュータを利用しようと考えた場合は、具体的には各情報処理教室のパソコンに向かうこととなります。これ以降に書かれている節を順々に読み進んでください。

### 2.5.1 パソコンから UNIX を利用する為の予備知識

ここではパソコンから UNIX コンピュータを遠隔操作で利用するための最低限の予備知識を説明します。あまり厳密に解釈しなくて結構ですので、イメージだけを頭に描いて読んでください。

<sup>5</sup> いつも授業で使っていて空気がない、部屋が遠い、階段を上るのが嫌、など

## 遠隔操作

遠隔操作と言いましたが、それほど不便でも面倒でもありません。身近な遠隔操作の例としてビデオデッキを挙げておきましょう。ビデオテープを見る際に、ビデオデッキの再生ボタンを押す場合もあるでしょうが、リモコン<sup>6</sup>の再生ボタンを押すときもあるでしょう。ビデオデッキに当たるのがUNIXコンピュータ、リモコンに相当するのがパソコンだと思ってください。つまりUNIXコンピュータに直接向かう代わりにパソコンからリモコン操作でUNIXコンピュータを利用しようということです。

ビデオデッキのリモコンは、デッキの方を向けて「ぴっ」とやってやるだけで良いのですが、コンピュータのリモコンは現在のところ電線でつないでやらないといけません。この役割りを果たす電線がすなわち京都産業大学のコンピュータ・ネットワーク<sup>7</sup>というわけです。ここでリモコンとして紹介するパソコンは全て予めネットワークにつながっていますから、UNIXコンピュータをパソコンから利用する場合にあなたがいちいち電線をつないだりする必要はありません。

## 用語について

UNIXコンピュータなどの為の遠隔操作機器はビデオデッキなどの場合と違って「リモコン」とは呼ばず、一般的には「端末」ないしは「ターミナル」と呼ばれます。端末を経由して利用しようとするコンピュータそのものはビデオデッキの場合と違って「本体」などとは呼ばず、一般的には「ホストコンピュータ」ないしは短く「ホスト」などと呼ばれます。UNIXコンピュータはほぼ例外なくホストコンピュータとなる為の機能を持っています。UNIXコンピュータ以外のコンピュータでもホストコンピュータとなる為の機能を持っているものは幾らもあります。

端末とホストコンピュータは常にネットワークで接続されていますが、ネットワーク上には複数のホストコンピュータが存在します。一般的にネットワークに接続された端末は、ホストコンピュータをどれにするか選ぶことができます。ちょうどリモコンを「ぴっ」とやる時に、向けた方向によって指示するビデオデッキを選べる<sup>8</sup>ような感じです。つまり端末を使い始める最初の時に、まずどのホストコンピュータを利用するか端末に指示してやるのです。この過程を「(端末をホストコンピュータに)接続する」と表現しています<sup>9</sup>。但し、実際にはcc環境で遠隔利用する場合のホストコンピュータはcc2000マシンになると思います。それ以外のマシンをホストに選ぶことはしないでください。これ以降の説明でもホストにはcc2000を利用することを前提に行なっています。

例えばあなたが情報処理教室のパソコンの前に座って遠隔操作でcc2000を利用している場合、「情報処理教室のパソコンを端末にcc2000をホストコンピュータとして利用している」などというように表現します。

## パソコンによるUNIX端末：telnet環境とX環境

パソコンはUNIXコンピュータとは別の独立したコンピュータですが、UNIXコンピュータを遠隔操作する機能を追加することができます。この「UNIXコンピュータのリモコンとなる為の機能を実現するもの」を端末ソフトないしはターミナル・エミュレータ・ソフトと呼んでいます。

端末ソフトが提供するUNIX環境は大別して二種類あります。文字情報だけを扱うtelnet<sup>10</sup>環境と、Xウィンドウによる文字情報とグラフィクス情報の両方を扱うX<sup>11</sup>環境です。様々な要因がある為、telnet

<sup>6</sup> Remote Controller、リモート・コントローラの略。即ち遠隔操作機器のことですね。

<sup>7</sup> このネットワークにはKING(「きんぐ」と読んでください)という名前が付けられています。

<sup>8</sup> 現実には余程広い部屋の四方の隅にビデオデッキを置かないとうまく行かないでしょうけど。

<sup>9</sup> これはネットワークなどというものが現れる以前には端末にホストを選ばせる事ができず、そうしたい場合は端末から出ている電線を実際に別のホストコンピュータにつなぎ変えていた歴史的経緯から来た表現だと思えます。

<sup>10</sup> 「てるねっと」と読んでください。

<sup>11</sup> 単に「えっくす」と読んでください。

環境と X 環境のどちらがあなたにとって使いやすいかは一概には言えません。いろいろ試してみるのが良いでしょう。

2.5 で遠隔操作として紹介した情報処理教室の全てのパソコンに telnet 端末ソフトが用意されています。

## 2.6 コンピュータウイルス

大変残念なことです。コンピュータを利用する上でコンピュータウイルスが非常な脅威となっています。パーソナルコンピュータを利用する上でもこれを避けて通ることはできません。特に計算機センターが管理しているパーソナルコンピュータにはマックと IBM PC 互換機が含まれており、これらは他の国産のパーソナルコンピュータに比べてウイルスの多さなどの理由からウイルス感染の危険性が非常に高いことを意識しておく必要があります。

パーソナルコンピュータに感染したウイルスが引き起こす害悪は様々です。例えば利用者の書類を瞬で消したり、書類の中の数字をほんの一桁だけ加算したりします。それは長時間の努力の積み重ねの仕事を消し去ったり、間違った結果をもたらします。ウイルスに感染することは被害者にとって悲しく腹立たしいことです。しかし、もっと残念なのは、自分が感染していることを知らずに人にもウイルスを移してしまうことです。

情報処理教室のパーソナルコンピュータはいつもウイルスの脅威にさらされています。いつ感染するか分からないし、いつあなたが被害を被るか分からないからです。このような理由から、ウイルスの被害を最小限にするため各情報処理教室への「外部ソフトウェア」の持ち込みは原則として認めておりません。

ここで言う「外部ソフトウェア」とは情報処理教室のパーソナルコンピュータに正式に認められていないソフトウェアを指しています。アプリケーション、システムソフトウェア等がこれに該当します。これらのソフトウェアの持ち込みがウイルス感染の原因となっているからです。

ウイルスからあなたの書類を守るために以下のことに気を配ってください。情報処理教室のパーソナルコンピュータにはウイルス除去用のソフトウェア (scan, Disinfectant 等) が入れてあります。

時々で良いのですが、そのウイルス除去ソフトウェアであなたがこれから使おうとするパーソナルコンピュータと、あなたのフロッピーを調べてください。このソフトウェアの動かし方が分からなければ、誰か知っていそうな人に尋ねるか、計算機センターメインカウンターで聞いてください。

もし人からフロッピーを預かって仕事をしなければならぬときは、そのフロッピーがコンピュータ・ウイルスに感染していないことを確かめてから仕事をする習慣を付けてください。

もし自分のフロッピーがコンピュータウイルスに感染していることが分かったら、面倒ですが計算機センターのメインカウンターに連絡してください。あなたのフロッピーからウイルスを除去するのは簡単ですが、むしろウイルスの感染経路を特定し、他にも感染している人がいないか調べる必要があるからです。

## 第3章 UNIXへの道

ここでは様々な場所の様々なコンピュータから cc 環境の UNIX マシンを利用する方法について、その部屋ごと、そのコンピュータごとに説明します。これまでの説明の部分であなたはもう既にどの部屋のコンピュータから login するか決めているはず<sup>1</sup>。

既に説明したように cc 環境の UNIX コンピュータを利用する場合、21 情報処理教室以外の部屋からはパソコンを利用して telnet もしくは X によって遠隔操作で cc2000 マシンを利用することになります<sup>2</sup>。

### 3.1 これ以降の読み方

これ以降は部屋ごと、コンピュータごとに説明を行いません。説明は節を分けて並列になっていますが順に読む必要はなく、あなたが利用しようと思っている部屋の部分についてだけ読めば良いでしょう。

- 10 号館情報処理教室の Netfinity(Linux) を使う (62 ページより)
- 10 号館情報処理教室の Netfinity(WindowsNT) を使う (50 ページより)
- 21 情報処理教室の Ultra1 を使う (34 ページより)
- 10206, 11, 32 情報処理教室及び図書館パソコン室の Macintosh から telnet で cc2000 を使う (42 ページより)
- 51 情報処理教室の DigitalPC200i 及び図書館パソコン室の Vectra から telnet で cc2000 を使う (50 ページより)

全ては例示を伴っていますので、この段階でとにかくその部屋に行って実際に作業しながら読み進んでいけばいいでしょう。各節ではその部屋に入ってコンピュータの電源を入れるところから login するまでと、そのコンピュータでの作業が終って logout して電源を切るまでの一連の操作を説明しています。実際に試してみて、一連の操作ができるようになったら 4 に進んでください。

---

<sup>1</sup> まだ決まっていなければあなたは重要なところを読み飛ばしています。今すぐ 15 ページの 2 まで戻ってください。

<sup>2</sup> telnet, X, 遠隔操作という単語に聞き覚えがなければやはりあなたは重要なところを読み飛ばしています。今すぐ 15 ページの 2 まで戻ってください。

## 3.2 Ultra1 を使う

Ultra1 は 2 号館 4 階の 21 情報処理教室に設置してあります。Ultra1 それ自身を利用するばかりでなく、計算機センターが用意している Enterprise3000 を利用する場合の窓口ともなるでしょう。

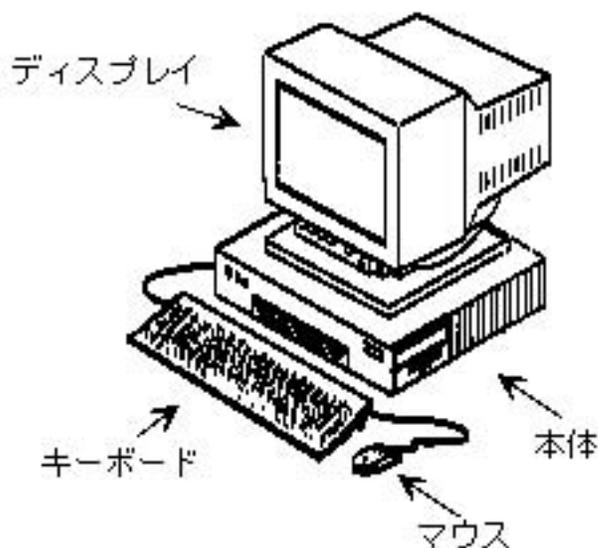


図 3.1 Ultra1 外観

### 3.2.1 まず部屋へ

Ultra1 は 2 号館 4 階の 21 情報処理教室に設置してあります。この部屋は常時カードロックシステムが働いており、学生証を入り口のドア横の機械に通すことによって鍵が開きます。入室可能な時間帯、曜日、期間に限りがありますので、付録の「情報処理教室の利用について」を一読してください。もしもカードを通して鍵が開かない場合は、カードの向き、表裏、カードを通過させる速度（遅すぎても速すぎても駄目）を変えてやってください。どうしても駄目な場合はカードの磁気情報がおかしくなっている可能性があります。カードを通す機械にあなたのカードでなぜ鍵が開かないか、その理由が出ていますので、それを控えて計算機センターまで連絡してください。

部屋に入るときは、そこで授業をやっていないことを確認しましょう。もし授業中であれば入室して利用していいかどうか指導教員に確認を取るのが礼儀というものです。部屋に誰も居なかった場合は照明、空調機が切れている場合があります。これらのスイッチは部屋に入ったそのすぐ右手の壁に集中して置いてありますので、自分で操作してください。部屋の管理そのものは理学部事務室が行っていますので、何か部屋の状態について質問、要望があれば理学部事務室までお願いします。コンピュータについての質問、要望は計算機センターのメインカウンターまでお願いします。

### 3.2.2 電源を入れる

自分が利用するコンピュータを選びます。これが最初の利用であれば、電源の入っていないマシンを選ぶのがいいでしょう。というのは、もし電源が入っていたら、それは誰かが既に使っているものかも知れ

ないからです。ディスプレイ（テレビの様な画面）の下方にボタンが並んでいますがその右端の長いボタンがディスプレイの電源ボタンです。

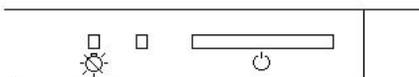


図 3.2 Ultra1 ディスプレイの電源

電源ボタンの左が使用中に点く緑のランプ、その左が本体の電源が切れたときの省電力モードを表すオレンジのランプです。緑、オレンジのどちらのランプも消えていたら電源ボタンを押してください。緑色のランプがどのくらい明るく点灯するか判らないので点灯しているのかいないのか区別が付かないという人は入り口に最も近いマシンのランプを見てください。このマシン (cssol000) は特別な仕事をしているため常時電源が入っていますので、比較になるでしょう。

ランプを確認せずに、画面が真っ暗だからと言ってそのマシンの電源が切れているかどうかは判りません。Ultra1 には、しばらく誰もキーボードを触らなかったら利用途中でも画面を真っ暗にしてしまう機能があるからです。念のためにマウスを動かしてみて、それでも画面に何の変化もないことを確認しましょう。電源が入っていないマシンが見つかったら、そのマシンのキーボードの右上にある電源キー（）を押してください。

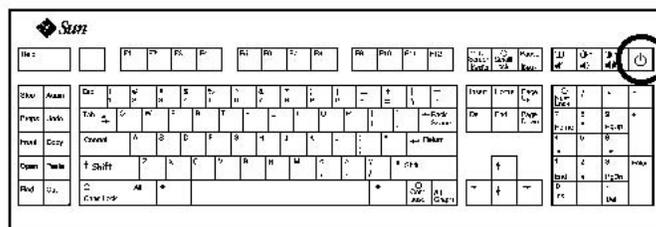


図 3.3 Ultra1 キーボード：電源スイッチ

電源が入ったらピーという音と共にファンが回る音がして、画面が白くなります。それから画面に色々な文字が流れていって、合計 2 分足らずで利用可能な状態になります。

### 3.2.3 login する

画面中央には以下のウィンドウが表示されていると思います。

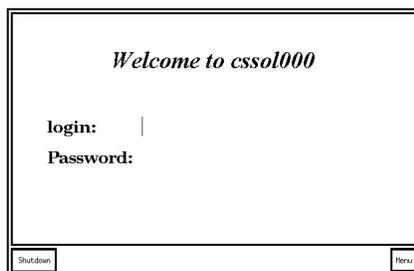


図 3.4 login ウィンドウ

キーボードを使ってまずあなたのユーザ ID をタイプします。ユーザ ID をタイプし終り、リターンキーを一度押せば今度はパスワードを入力できる状態になります。ここでパスワードを入力するのですが、パスワードは横から見ている他の人にバレないように、タイプしてもその文字が画面に表示されません。キーを打ち間違えないように注意して入力し、もう一度リターンキーを押します。うまくユーザ ID とパスワードを入力できれば画面表示が変わります。キーを打ち間違えたりして login に失敗するとウィンドウの下の部分に赤い文字で「Login incorrect」と表示されます。この場合はもう一度ユーザ ID のタイプからやり直します。

もしこれがあなたにとって Sign 登録後初めての login であればパスワードは Sign 登録時にあなたがメモしたもののはずです。もし何度試しても login できないようであればユーザ ID、パスワードのいずれかが間違っているか、まだあなたのユーザ登録作業が計算機センター側で完了していないのです。計算機センターに自分のユーザ ID が既に登録されているかどうか確認してください。もしもユーザ ID も正しく、また登録作業も完了しているなら、これはパスワードが違っているとしか考えられません。計算機センターメインカウンターに連絡してパスワードを強制変更して貰ってください。

### 3.2.4 ちょっとメッセージ

login すると、ときには利用者宛の通知、おしらせが以下のように画面に表示されるかもしれません。

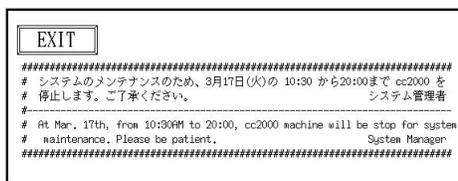


図 3.5 login 直後のメッセージ

よく読んで、自分に関係があるかどうか確認してください。確認が済めば、リターンキーを一度押すだけでこの表示を消すことができます。もしも内容に付いてよく判らないことがあれば計算機センターメインカウンターまで連絡するのがいいでしょう。

### 3.2.5 ターミナルはあるかな？

login に成功すると以下のような画面表示になっていると思います。

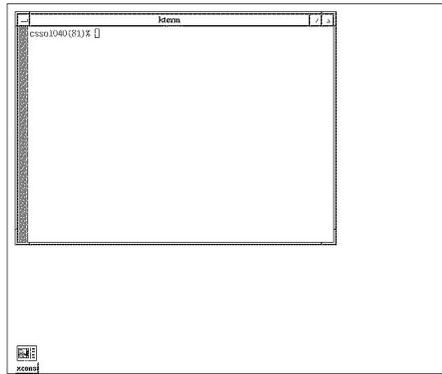


図 3.6 X の初期画面

画面の中に以下の様なウインドウがあるかどうかさがしてください。そのようなウインドウが見つければ、ひとつ飛ばして「ターミナルは大丈夫かな？」まで進んでください。もしも見つからなければこのまま次に進んでください。

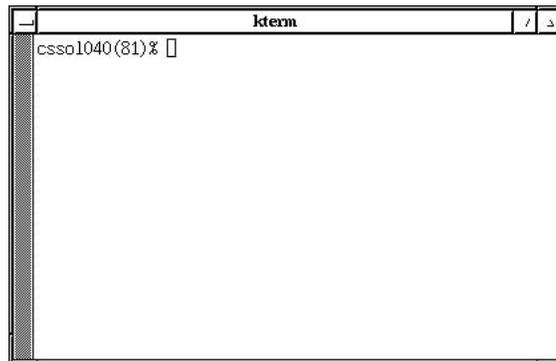


図 3.7 KTerm

#### ターミナルを起動する

画面の背景、特にウインドウの無い場所にマウスカーソルを持って行って、マウスの左ボタンを押せばなしにします。すると以下のようなメニューが現れるでしょう。この一連の操作を「ルートメニューを表示させる」と呼びます。

Root Menu
KTerm
Mule
Netscape
Applications ▾
MO Disk ▾
Floppy Disk ▾
Restart fvwm
Exit

図 3.8 ルートメニュー

マウスのボタンを押したままにして、マウスカーソルを「KTerm」と書いてある行の上に持っていきます。「KTerm」の行の上まで来たらその場所でマウスボタンを離します。これで KTerm が起動されるはずですが。

この一連の操作を「メニューから「KTerm」を選択する」と呼びます。このメニューでは「KTerm」が選択されると KTerm が起動するようになっています。

ターミナルは大丈夫かな？

ターミナルのウィンドウの左隅には `cssol001(81)%` などと表示されていると思います。(但し数字は違うかもしれません。)ここで試しに `date` とタイプしてリターンキーを押してみてください。文字をタイプしても `cssol001(81)%` に続いて表示されない場合は、ターミナルウィンドウがアクティブになっていません。一度 `cssol001(81)%` 辺りをクリックしてターミナルウィンドウをアクティブウィンドウにした後、`date` とタイプし直してください。以下のように今日の日付と時間が表示されれば大丈夫、ターミナルを使える状態になっています。

```
cssol001(81)% date
1997年03月25日(火)19時28分49秒 JST
cssol001(82)%
```

### 3.2.6 パスワードを変える

もしもこれがあなたにとって Sign 登録後初めての login ならば、ここでまずパスワードを変更しましょう。Sign 登録時のパスワードは機械的に決められたもので、余り安全とは言えません。必ずパスワードをあなた自身が選んだ言葉に変更してください。パスワードを決める際には 19 ページの 2.1.10 で既に述べた事に注意してください。また、コンピュータにはパスワード付け替えの際に安全なパスワードかどうか、簡単なチェックをする機能がついています。メッセージの指示に従ってください。

パスワードを変更するには `yppasswd` コマンド<sup>3</sup> を利用します。上の節で説明したターミナルを見ると、

```
cc2000(81)%
```

<sup>3</sup> 綴りがちょっと変ですが、間違っていないです。

などとなっていますね。(但し cc2000 という部分や括弧内の数字は例と違うかもしれません。)ここで yppasswd とタイプし、リターンキーを押します。すると以下のような状態に成りますね。(以下はユーザ ID ozaki の例。ozaki の部分にはあなたのユーザ ID が表示されているはずです。)

```
cc2000(81)% yppasswd
yppasswd: Changing password for ozaki
Enter login(NIS) password:
```

指示通りここでは古い、つまり今先ほどまで使っていたパスワードをタイプし、リターンキーを押します。ここでは login の時と同じくパスワードをタイプしている間、横の人の覗き見によってあなたのパスワードがバレないように、タイプした文字は画面に表示されない事に注意してください。表示されなくともタイプした文字はちゃんと入力されていますから、安心して確実に一文字ずつタイプして最後にリターンキーをタイプしてください。すると今度は以下のようなプロンプトを表示します。

```
New password:
```

では指示通り新しい、つまり次からこれにしようというパスワードをタイプし、リターンキーを押してください。ここでもタイプした文字は表示されません。今度は以下のようなプロンプトを表示します。

```
Re-enter new password:
```

これはいましがたタイプした新しいパスワードが打ち間違いで無いことを確認するためのものです。もう一度新しいパスワードをタイプしてリターンキーを押してください。今度は以下のような状態になります。

```
NIS(YP) passwd/attributes changed on ccnic
cc2000(82)%
```

この NIS (YP) passwd/attributes changed が表示されれば成功です。もしもパスワードが短すぎたりタイプミスがあったりしたらなんらかのエラーメッセージが表示されます。途中でおかしくなったと思う場合は C-c (つまりコントロールキーを押しながら C キーを押す) で cc2000(82)% を表示させます。それからもう一度 yppasswd コマンドをやり直します。

よく表示されるエラーメッセージを紹介しておきます。

```
Password must contain at least two alphabetic characters and
at least one numeric or special character.
```

パスワードはアルファベットと数字や記号を混ぜたものにしてください。アルファベットや数字だけのものは推測されやすく、危険です。

```
Password is too short - must be at least 6 characters.
```

パスワードは 6 文字から 8 文字にしてください。(9 文字以上入力することもできますがコンピュータが判別するのは 8 文字目までです。)

```
Passwords must differ by at least 3 positions
```

古いパスワードと変化の少ないものを付けることはできません。

```
Too many failures - try later.
```

失敗しすぎです。もう一度最初からどうぞ。

パーミッションが与えられていません。

何かを失敗したためにユーザ本人と認められませんでした。そのためパーミッション（権限）がないと判断されました。おそらく古いパスワードを間違えて入力したのでしょう。

### 3.2.7 logout する（セッションを終了する）

他にやりたいことも色々あるでしょうが今回は最初の利用でしょうし、まず一通りの操作をやってみるという意味で、logout をこの場でやってみることを勧めます。

画面の背景、特にウインドウの無い場所にマウスカーソルを持って行って、マウスの左ボタンを押しっぱなしにします。すると以下のようなメニューが現れるでしょう。この一連の操作を「ルートメニューを表示させる」と呼びます。

Root Menu
KTerm
Mule
Netscape
Applications ▾
MO Disk ▾
Floppy Disk ▾
Restart fvwm
Exit

図 3.9 ルートメニュー

マウスのボタンを押したままにして、マウスカーソルを恐らくは一番下にある「Exit」と書いてある行の上を持っていきます。「Exit」の行の上まで来たらその場所でマウスボタンを離します。この一連の操作を「メニューから「Exit」を選択する」と呼びます。このメニューでは「Exit」が選択されると logout（セッションの終了）作業が行なわれるようになっています。Exit すると元の login 画面に戻ります。

これが Ultra1 での logout 操作です。実際には logout という名前のコマンドも操作も実行しないことに注意してください。Ultra1 ではつまり logout とはセッションの終了だということです。セッション終了の操作を行なって数十秒すると、最初に電源を入れて login 操作をする直前の状態に戻って図 3.4 が表示されます。もしもその状態にならない、つまり正しくセッションが終了できなければ、計算機センターメインカウンターまで連絡ください。そのまま放置するのはいけません。また電源を無理に切るのはコンピュータが壊れるので厳禁です。

### 3.2.8 シャットダウンする

セッションが終了できれば、最初に電源を入れて login する直前の状態に戻ります。

#### 注意

本体とキーボードの間に「このマシンの本体電源は切らないでください」というようなカードが置いてあった場合は、この節は読み飛ばして 3.2.9 に進んでください。恐らくそのマシンは誰かほかの人が夜間

利用するつもりなのです。

今度は図 3.4 の左下隅のボタンに注目してください。

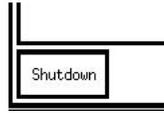


図 3.10 Shutdown ボタン

ここで今度はこのボタンをクリックしてください。すると数十秒で画面が白くなって、電源が自動的に切れます。この操作をシャットダウンと呼んでいます。

電源が自動的に切れない場合は何かトラブルがあったと思われます。決して電源スイッチを切らないでください。また、おかしなまま放置するのもいけません。計算機センターメインカウンターまで連絡ください。また、部屋の入り口に最も近い場所にある cssol000 と、cssol000 正面に向かって右隣にある cssol001 の二台については電源は常時入りっぱなしです。このマシンはシャットダウンできないようになっています。この二台は電源を切らないでください。

### 3.2.9 部屋を出る

マシンの電源を切って部屋を出るとき、他に使用者がいなければ空調機、照明も切って行きましょう。

### 3.2.10 さて、さて、

最初の利用はこれでおしまいです。大変よくできました。あとは精進して UNIX ユーザへの道を歩まれるわけですね。それには UNIX での一般的なコマンド操作などについて説明している、4 へと進んでください。

### 3.2.11 マニュアルなど

Ultra1 のマニュアルは部屋の入り口から最も離れた奥のロッカーに数冊置いてあります。但し一部ずつしかないので皆で同時に見ることはできません。うまく共有してください。

次は、71 ページの 4 へ進んでください。

### 3.3 Macintosh の telnet で cc2000 を使う

ここでは1号館2階の11情報処理教室、3号館2階の31情報処理教室、3号館1階の32情報処理教室、10号館2階の10206情報処理教室及び図書館1階のパソコン室のMacintoshからtelnet環境でcc2000をホストコンピュータとして利用する手続きについて説明します。

Macintosh(以下「マック」という<sup>4</sup>。)はApple Computer社製のパソコンで、学内には、1号館2階の11情報処理教室にPowerMacintosh 6100/66という機種が90台、3号館2階の31情報処理教室にPowerMacintosh G3 MT300という機種が44台、3号館1階の32情報処理教室にPowerMacintosh 8500/180という機種が96台、10号館2階の10206情報処理教室にPowerMacintosh G3 MT300という機種が48台、図書館1階のパソコン室にPowerMacintosh 6100/66という機種が12台とPowerMacintosh 7200/120が12台、計302台設置してあります。

これらの部屋のマックでは端末ソフトとして「BetterTelnet」を利用します。このソフトはマックでtelnet環境を提供します。では、さっそく使ってみましょう。

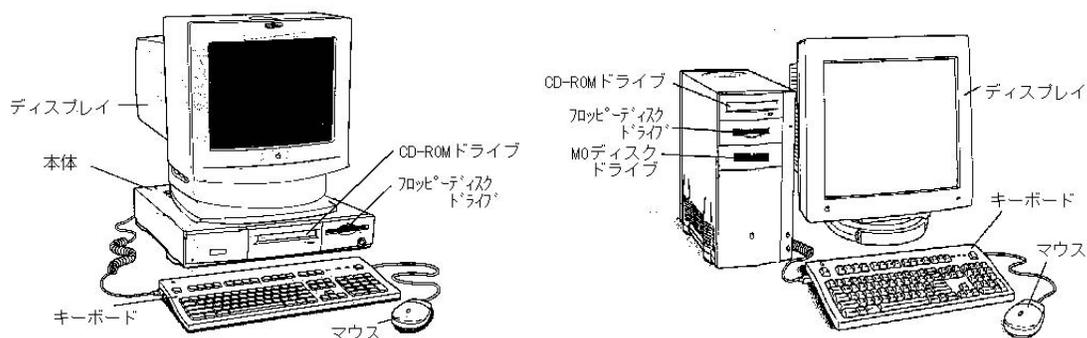


図 3.11 Power Macintosh 外観

#### 3.3.1 まず部屋へ

31,32,10206情報処理教室は常時カードロックシステムが働いており、学生証を入り口のドア横の機械に通すことによって鍵が開きます。入室可能な時間帯、曜日、期間に限りがありますので、付録の「情報処理教室の利用について」を一読してください。もしもカードを通して鍵が開かない場合は、カードの向き、表裏、カードを通過させる速度(遅すぎても速すぎても駄目)を変えてやってください。どうしても駄目な場合はカードの磁気情報がおかしくなっている可能性があります。カードを通す機械にあなたのカードでなぜ鍵が開かないか、その理由が出ていますので、それを控えてもよりの事務室まで連絡してください。

11情報処理教室は利用可能な時間には鍵が開いています。入室可能な時間帯、曜日、期間に限定がありますので、やはりこちらも付録の「情報処理教室の利用について」を一読してください(教室前廊下にある掲示板も参照のこと)。図書館パソコン室は図書館開館時間中に使うことができます。

部屋に入るときは、そこで授業をやっていないことを確認しましょう。もし授業中であれば入室して利用していいかどうか指導教員に確認を取るのが礼儀というものです。部屋に誰も居なかった場合は照明、空調機が切れている場合があります。部屋の入り口近くの壁にスイッチ、またはリモコンがありますので、自分で適当に操作してください。部屋によってはスイッチに触れないようお願いしています。その場合はスイッチ近くに掲示がされていますのでそれに従ってください。部屋の管理そのものは12ページで紹介

<sup>4</sup> Macintosh ユーザーは Macintosh のことを、親しみを込めてこう呼びます。

した所管事務室が行っていますので、何か部屋の状態について質問、要望があればそちらまでお願いします。コンピュータについての質問、要望は情報処理教室については計算機センターのメインカウンターまで、図書館パソコン室については図書館のメインカウンターまでお願いします。

### 3.3.2 電源を入れる

今から自分が利用するコンピュータを選びます。もしこれが最初の利用であれば、電源の入っていないマックを選ぶのがいいでしょう。というのは、もし電源が入っていたら、それは誰かが既に使っているかも知れないからです。誰かが使っているマックはディスプレイ（テレビの様な画面）が明るく、画面の右下の緑色のランプが点灯しているので識別できます。

では、電源を入れます。マックのキーボードを見てください。キーボード上部に孤立した長四角のボタン（三角の印が入っています）があります。これが電源スイッチです。押してみましょう。

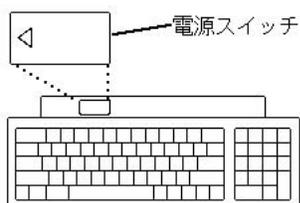


図 3.12 電源スイッチを押して電源を入れる

本体の電源を入れるとディスプレイも連動して電源が入ります。やがてファンが回る音がして「ポワーン」と音がします。それからディスプレイ画面が明るくなり、画面の下に色々なマークが表示されて、2分足らずで利用可能な状態になります。

### 3.3.3 マックの操作

せっかくマックの電源が入って使えるようになったのですから、少しマックの操作を学びましょう。マックの中には「Macintoshの手引き」というソフトウェアが入っています。これは、マックの基本的な使い方が全て解り、約30分ほどでできるので、是非試してください。

### 3.3.4 端末ソフトの起動

まず端末ソフトを起動しましょう。マックのマウスを使ってアップルメニューから Telnet cc2000 を選びます。うまく起動できるとウインドウが一つ開きます。

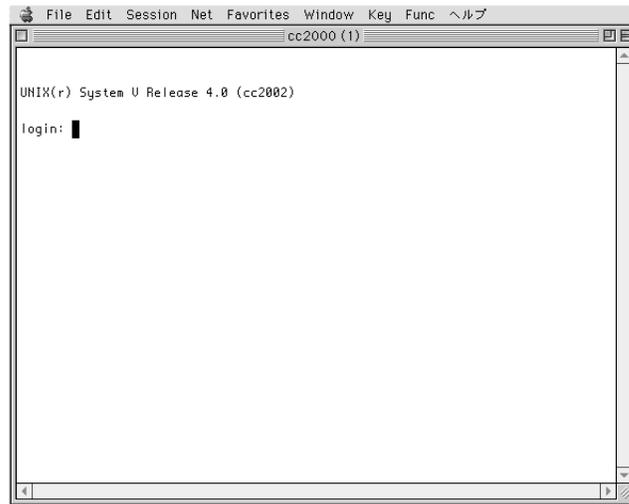


図 3.13 Bettertelnet の login 画面

login: と催促文字が表示されているはずですが。

### 3.3.5 さあ、login しよう

いよいよ login です。ここからはもうあなたのキーボードに対する入力はホストコンピュータに対する遠隔操作となっています。

ユーザ ID とパスワードを入力しましょう。

正しくユーザ ID をタイプし<Return>キーを押すと、パスワードを聞いて来ます。これも正しくタイプし<Return>キーを押すと login できます。入力するパスワードは横から見ている人にばれないように、タイプしても画面に表示されないの注意して正しく入れてください。

もしパスワードの入力途中で間違ったときは、それまで入力した文字数より少し多めに<Delete>キーを押してもう一度最初からパスワードを正しく入れてみましょう。

ちゃんと login できましたか？

いくらユーザ ID とパスワードを入れても login できない方は、ユーザ ID とパスワードを間違っ入れていませんか。UNIX は大文字と小文字の区別があるので注意してください。

どうしてもうまくいかない場合はもよりの補助員まで連絡してください。

### 3.3.6 ちょっとメッセージ

login すると、ときには利用者宛の通知、おしらせが以下のように画面に表示されるかもしれません。

```
Sun Microsystems Inc. SunOS 5.3 Generic September 1993
4月15日深夜、ファイルサーバである ccnfs (cc2000) の
レポートが予定されています。
これにともない cc 環境の全てのマシンの利用が出来なくな
ります。夜間に長時間のジョブを実行する予定のある利用者
はそれまでに実行が終るよう注意して下さい。
cc2000(84) %
```

図 3.14 login 直後のメッセージ

よく読んで、自分に関係があるかどうか確認してください。もしも内容に付いてよく判らないことがあれば計算機センターメインカウンターまで連絡するのがいいでしょう。

### 3.3.7 パスワードを変える

もしもこれがあなたにとって Sign 登録後初めての login ならば、ここでまずパスワードを変更しましょう。Sign 登録時のパスワードは機械的に決められたもので、余り安全とは言えません。必ずパスワードをあなた自身が選んだ言葉に変更してください。パスワードを決める際には 19 ページの 2.1.10 で既に述べた事に注意してください。また、コンピュータにはパスワード付け替えの際に安全なパスワードかどうか、簡単なチェックをする機能がついています。メッセージの指示に従ってください。

パスワードを変更するには `yppasswd` コマンド<sup>5</sup> を利用します。上の節で説明したターミナルを見ると、  
cc2000(81)%

などとなっていますね。(但し `cc2000` という部分や括弧内の数字は例と違うかもしれませんが。)ここで `yppasswd` とタイプし、リターンキーを押します。すると以下のような状態に成りますね。(以下はユーザ ID `ozaki` の例。`ozaki` の部分にはあなたのユーザ ID が表示されているはずです。)

```
cc2000(81)% yppasswd
yppasswd: Changing password for ozaki
Enter login(NIS) password:
```

指示通りここでは古い、つまり今先ほどまで使っていたパスワードをタイプし、リターンキーを押します。ここでは login の時と同じくパスワードをタイプしている間、横の人の覗き見によってあなたのパスワードがバレないように、タイプした文字は画面に表示されない事に注意してください。表示されなくともタイプした文字はちゃんと入力されていますから、安心して確実に一文字ずつタイプして最後にリターンキーをタイプしてください。すると今度は以下のようなプロンプトを表示します。

New password:

では指示通り新しい、つまり次からこれにしようというパスワードをタイプし、リターンキーを押してください。ここでもタイプした文字は表示されません。今度は以下のようなプロンプトを表示します。

Re-enter new password:

これはいましがたタイプした新しいパスワードが打ち間違いで無いことを確認するためのものです。もう一度新しいパスワードをタイプしてリターンキーを押してください。今度は以下のような状態になります。

<sup>5</sup> 綴りがちょっと変ですが、間違っはしません。

```
NIS(YP) passwd/attributes changed on ccnic
cc2000(82)%
```

この NIS (YP) passwd/attributes changed が表示されれば成功です。もしもパスワードが短すぎたりタイプミスがあったりしたらなんらかのエラーメッセージが表示されます。途中でおかしくなったと思う場合は C-c (つまりコントロールキーを押しながらCキーを押す) で cc2000(82)% を表示させます。それからもう一度 yppasswd コマンドをやり直します。

よく表示されるエラーメッセージを紹介しておきます。

```
Password must contain at least two alphabetic characters and
at least one numeric or special character.
```

パスワードはアルファベットと数字や記号を混ぜたものにしてください。アルファベットや数字だけのものは推測されやすく、危険です。

```
Password is too short - must be at least 6 characters.
```

パスワードは6文字から8文字にしてください。(9文字以上入力することもできませんがコンピュータが判別するのは8文字目までです。)

```
Passwords must differ by at least 3 positions
```

古いパスワードと変化の少ないものを付けることはできません。

```
Too many failures - try later.
```

失敗しすぎです。もう一度最初からどうぞ。

パーミッションが与えられていません。

何かを失敗したためにユーザ本人と認められませんでした。そのためパーミッション(権限)がないと判断されました。おそらく古いパスワードを間違えて入力したのでしょうか。

### 3.3.8 logout をしよう

次は、logout です。他にやりたいことも色々あるでしょうが今回は最初の利用でしょうし、まず一通りの操作をやってみるという意味で、logout をこの場でやってみることを勧めます。

画面上にcc2000(81)%が表示されていることを確認してください。ただし、括弧の中の数字は気にしないでね!!

ここで、logout<Return>と入力するだけで telnet ウィンドウが消え logout されます。

```
cc2000(81)% logout
```

### 3.3.9 端末ソフトを終了する

logout するといきなり telnet ウィンドウが消えます。少し驚くかもしれませんが、それでいいのです。すでにホストコンピュータとの遠隔操作を終わっているのです。

さて、端末ソフトを終了するにはマウスで「File」メニューの「Quit」を選択します。これで BetterTelnet は終了します。

さあ、どうでしたか？スムーズにここまで来れたでしょうか。すこしややこしかったかもしれませんが、決して難しくはなかったと思います。はじめは誰もが初心者です。くじけないで頑張れば、すぐに鼻歌まじりで使えるようになります。

### 3.3.10 電源を切る

さて、マックに慣れて使えるようになったのですが、ここでマックの電源を切って止める操作をしてみましょう。

普通、パソコンの電源を切るときは、いま使っていた機能を終了する必要があります。ワープロならワープロソフトを、通信なら通信ソフトを終了してから行いますが、ここではまだ何も使っていないので気にする必要はありません。

画面上部の「特別」メニューより「システム終了」を選びます。選ぶ操作は先ほど「Macintosh の手引き」で練習しましたが、少しおさらいをしながら行なってみます。まず、マウスカーソルを「特別」メニューに移動させマウスボタンを押したままにします。そうすると、メニューの内容が下方向にプルダウン方式で表示されます。マウスボタンを押したままマウスカーソルを「システム終了」までドラッグして「システム終了」命令が黒く反転し強調されてからマウスボタンを放します。しばらくするとマック本体とディスプレイの電源が自動的に切れます。

マック本体とディスプレイの電源スイッチには手を触れないでください。

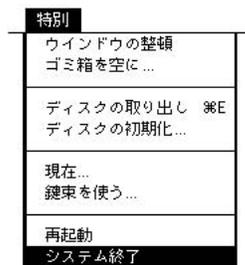


図 3.15 「特別」メニューから「システム終了」を選ぶ

### 3.3.11 困った時の対処

「システム終了」を選択しても終了されない

通常、「システム終了」が選ばれると、そのとき起動されているソフトウェアは自動的に終了していきませんが、一部この手順をうまく行なえないソフトウェアも存在します。なにか終了し忘れていたソフトウェアがないか確認し、もしあれば終了してください。うまくいけばシステムも終了されます。

文字をタイプしても表示されない

目的の telnet ウィンドウがアクティブになっていないことが考えられます。アクティブになっているかいないかはウィンドウのタイトルバーに横線が現れているかいないかで知ることができます。

入力したいウィンドウの一部をマウスボタンでクリックするとそのウィンドウがアクティブになります。

文字をタイプすると日本語が出てしまう

キー入力が日本語モードになっていることが考えられます。

日本語入力モードをやめて英語入力モードにするには command キーを押しながら Space バー (空白キー) を押します。メニューバーの右から二つ目の小さなアイコンが「米国旗」になれば英語入力モードになっています。

telnet 画面が表示されない

最初にアップルメニューから Telnet cc2000 を選択した時に telnet 画面が現れないなんてことは、あなたの操作が正しく、かつ cc2000 が稼働しておれば考えられないことです。

このような場合は近くで使っている人が cc2000 をうまく利用できているかどうか調べましょう。というのもホストコンピュータである cc2000 そのものが何らかの事情で止まっているとか、ネットワークが不通になっていることが考えられるからです。もし近くの人にも cc2000 との接続がうまく行っていないようなら、計算機センターメインカウンターに電話をしてください。

日本語が壊れたようになって正しく表示されない

これもアップルメニューから Telnet cc2000 を選択して使っているときは起こらない現象です。一旦 logout して、アップルメニューから Telnet cc2000 を選択する操作からやり直してください。

### 3.3.12 cc2000 以外のホストコンピュータを使う時の注意

BetterTelnet を使って cc2000 以外のホストコンピュータに login する事もできます。BetterTelenet アイコンを直接ダブルクリックして BetterTelnet を起動するか、何とかして BetterTelnet が起動されている状態で、File メニューの Open Connection... を選択する事によってホストコンピュータを指定する為のウィンドウが表示されます。ここでホスト名をタイプして<Return>キーを押せば指定したホストコンピュータへの login が可能になるでしょう。

ところでアップルメニューの Telnet cc2000 を選択して使っている時は計算機センターが用意した各種の設定が効いているので正しく動作していますが、Open Connection... でホストコンピュータを指定した場合はそれらの設定が及んでいけませんので幾らかの動作でおかしくなる可能性があります。以下に典型的な症状と原因、対処法を示しておきます。

telnet ウィンドウが表示されない

Open Connection... ウィンドウでタイプしたホストコンピュータの名前を (例えば cc2000 ならば cc200 など) 間違っている可能性があります。File メニューの Open Connection... を選択する操作からやり直してください。

ホスト名を間違えていない場合には、目的のコンピュータが何らかの事情で止まっているとか、ネットワークが不通になっている可能性があります。目的のコンピュータの管理者などに確認してみてください。

#### 漢字の部分が正しく表示されない

BetterTelnet がホストコンピュータに送る日本語コードとホストコンピュータから送られてくる漢字コードがうまく合っていないときにこの様な状況になります。(この表現の意味が判らない場合は cc2000 以外のホストコンピュータを利用しない方がいいでしょう。)

「Session」メニューに「Translation」または「Code」という項目があります。自分が使おうと思っているホストコンピュータの漢字コードが何かを確認して、これを正しい値(例えば cc2000 ならそれぞれ「EUC」または「EUC & JIS7」)にしてください。

これで日本語が正しく表示されます。

### 3.3.13 マニュアルなど

マニュアルは情報処理教室やパソコン室のロッカーか書架に置いてあります(マニュアルを設置していない部屋もあります)。使っているマックのそばに持ってきて利用して構いませんが、終わればみんなのために元に戻しておいてください。

次は、71 ページの 4 へ進んでください。

## 3.4 WindowsNT の telnet から cc2000 を使う

ここでは 5 号館 1 階の 51 情報処理教室にある Digital PC200i、図書館パソコン室にある HP Vectra VL 6/300、10 号館情報処理教室（10206 情報処理教室を除く）にある IBM Netfinity 3000 から telnet 環境で cc2000 をホストコンピュータとして利用する手続きについて説明します。なお、10 号館情報処理教室では、1 台のパソコンから 2 つの OS が選択ができます。ここでは Microsoft 社の WindowsNT4.0 について説明します。

これらのパソコンには、Microsoft 社の WindowsNT4.0 がインストールされており、15 ページ、2 で説明したとおり、まずユーザー確認が行なわれます。このユーザー確認は、cc 環境のユーザー名と同じユーザー ID を使用していますが、パスワードは違うものですのでご注意ください。

### 3.4.1 まず部屋へ

情報処理教室は常時カードロックシステムが働いており、学生証を入り口横の機械に通すことによって鍵が開きます。入室可能な時間帯、曜日、期間に限りがありますので、付録の「情報処理教室の利用について」を一読してください。もしカードを通して鍵が開かない場合は、カードの向き、表裏、カードを通過させる速度（遅すぎても速すぎても駄目）を変えてください。どうしても駄目な場合はカードの磁気情報がおかしくなっている可能性があります。10 号館 3 階の計算機センターのメインカウンターまで連絡してください。

図書館パソコン室は開館時間にご利用頂けます。

入室した際、部屋に誰も居なかった場合は照明が切れている場合があります。照明のスイッチはドア付近の左手または右手の壁にあります。自分で探し操作してください。

部屋の状態、コンピュータについての質問、要望があれば計算機センターまでお願いします。

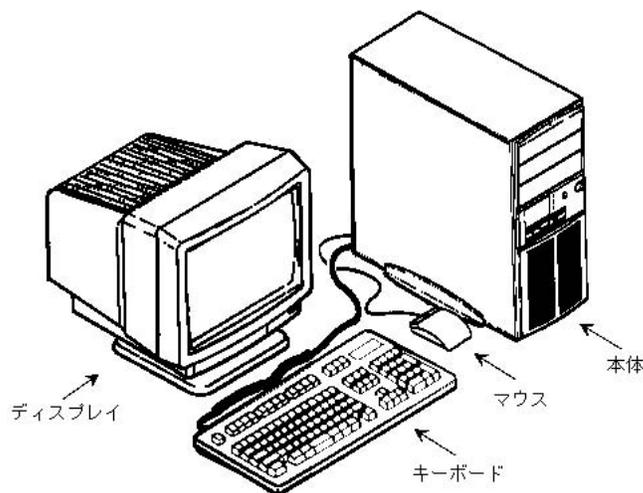


図 3.16 コンピュータ 外観

### 3.4.2 電源を入れる

自分が利用するコンピュータを選びます。この場合、電源の入っていないパソコンを選ぶのがいいでしょう。というのは、もし電源が入っていたら、それは誰かが既に使っているかも知れないからです。電源が入っていないパソコンは、ディスプレイや本体に緑色のランプが点灯していないことで識別できます。ランプを確認せずに、画面が真っ暗だからと言ってそのパソコンの電源が切れているかどうかは判りませんので、ディスプレイと本体の両方を必ず確認しましょう。

電源が入っていないパソコンが見つかったら、本体の電源スイッチを押してください。正しく押せると緑色のランプが点灯し、電源が入ったことが確認できます。

10号館情報処理教室では、図3.17が表示されます。キーボード上の矢印キー「↑」「↓」を押して、「Windows NT Workstation」を黒く反転させてください。10号館以外の情報処理教室では、そのままお待ち下さい。



図 3.17 10号館情報処理教室の起動画面

画面上に「ログオンの開始」と書かれたウインドウが表示されると利用可能な状態になります。それまでの間、何も触らず待っててください。



図 3.18 起動後の画面

### 3.4.3 WindowsNT にログオンしましょう

UNIX コンピュータを利用するには login と呼ばれる確認作業があることは既に述べたとおりですが、同様に WindowsNT を利用する時に行う確認作業をログオンと呼びます。コンピュータの電源を入れるのに成功すると画面の色が青色になり、しばらくすると「ログオンの開始」と書かれたウインドウが画面に表示されます。そこには、「ログオンするには、Ctrl+Alt+Del キーを押してください」と書かれてあります。これは、「Ctrl」キーと「Alt」キーと「Delete」キーを同時に押すことを意味します。この三つのキーを同時に押してください。うまく押せると「ログオン情報」のウインドウが現れ、ユーザー名、パスワードが入力できます。うまくいかないときは「Ctrl」キーと「Alt」キーを両方押しっぱなしにして「Delete」キーを押してみてください。



図 3.19 ログオン画面

では、あなたの cc 環境の Windows NT のユーザー名<sup>6</sup> とパスワード<sup>7</sup> を入れてください。具体的にはユーザ ID を入力して Tab キーもしくはマウスでパスワード欄をクリックしてカーソルをパスワード欄に移し、パスワードを入力してリターンキーを押してください。パスワードは他人に見えないように「\*」の記号で表示されます。正しく入力できていると信じてリターンキーを押せば大丈夫です。初期パスワードは、教室設置の「情報処理教室ガイド」を参照してください。

キーボードでは、一般的に「Enter」と呼ばれるキー<sup>8</sup>には、折れ曲がった矢印記号(↵)が刻印されています。ここではこのキーを「リターンキー」と表現します。文中にリターンキーを押すように書かれている場合は、この折れ曲がった矢印記号(↵)キーを押してください。

ログオンした後、あなたのパスワードが初期設定のままなら、すぐにパスワードを変更してください。変更方法は、次章を参照してください。ログオンに成功すると、図 3.20 の画面になります。

ログオンの途中で図 3.21 が表示されたら、「無視」または「中止」を選択してください。なお、利用に際して問題はありません。

### 3.4.4 まずは WindowsNT のパスワードを変更しましょう

すでに WindowsNT のパスワードを変更した方は、次章の 3.4.5 に進みましょう。また、定期的にパスワードを変更する心がけてください。パスワードの変更がまだなら、この章を読んで変更してください。

WindowsNT と UNIX の 2 つのパスワードを管理するのが面倒な方は、どちらか一方のパスワードに合わせましょう。どちらかのパスワードを変更した方は、もう一方のパスワードも同

<sup>6</sup> cc 環境の UNIX のユーザー名と同じです。

<sup>7</sup> cc 環境の UNIX のパスワードとは別です。

<sup>8</sup> キーボード真中やや右の大きめのキー

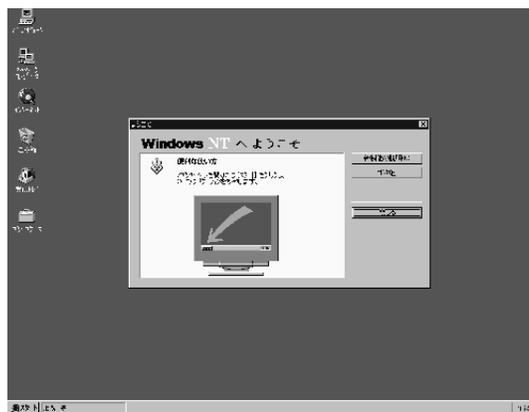


図 3.20 ログオン完了後の画面

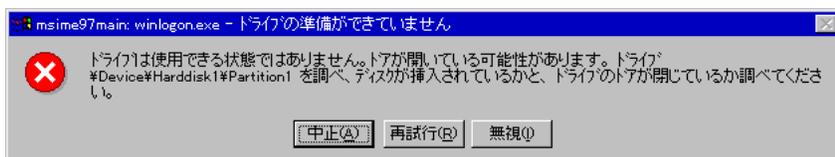


図 3.21 ログオン途中のエラー画面

じように変更すると便利です。なお、パスワードはあなたにとってキャッシュカードの暗証番号と同じようなものです。扱いに十分注意してください。

図 3.20 の状態になれば、ログオンのウインドウを出したときと同じようにCtrl キーとAlt キーとDelete キーを同時に押してください。

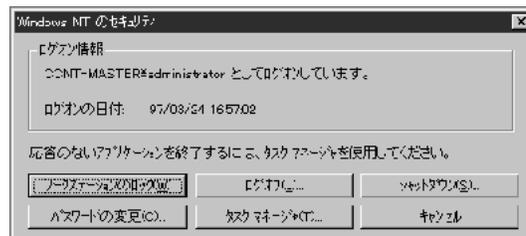


図 3.22 パスワード変更選択画面

このようなウィンドウが現れたら「パスワード変更」を選択してください。



図 3.23 パスワード変更画面

パスワード変更のウィンドウが表示されます。古いパスワード欄に今のパスワードを入力し、新しいパスワードを入力します。確認のためもう一度入力するボックスがあります。そこにも新しいパスワードを入力してください。

なお、入力した文字はセキュリティの関係で表示されません。慎重に入力して、間違いのないようなら「OK」を押してください。これで WindowsNT のパスワードが変更されます。

### 3.4.5 端末ソフトを起動しましょう

ここでは 5 号館 1 階の 51 情報処理教室の Digital PC200i から telnet 環境で cc2000 をホストコンピュータとして利用する手続きについて説明します。他の情報処理教室、図書館パソコン室でもほとんど同じ手続きで利用ができますので、参考にしてください。端末ソフトとしての「Tera term PRO」を利用します。

画面左側に「共通アプリケーション」と名前のついたアイコンがありますね。マウスポインタをそのアイコンに重ね合わせ、マウスの左ボタンをダブルクリックすると、ウィンドウが現れます。そのウィンドウ内に「TeraTermPRO」と名前のついたアイコンがありますね？マウスポインタを重ね合わせ、マウスの左ボタンをダブルクリックしてください。「TeraTermPRO」が起動します。

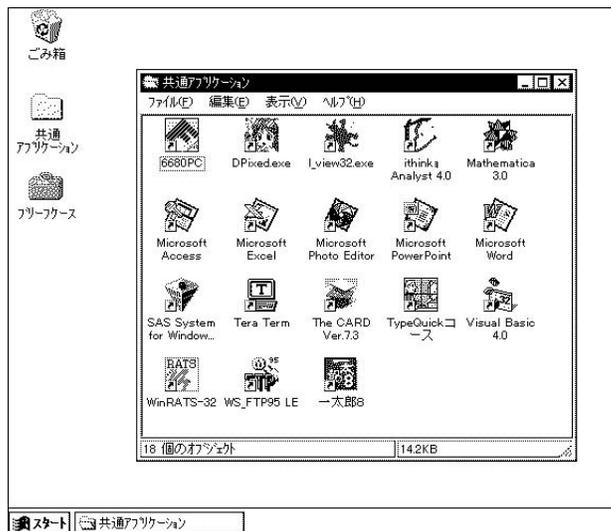


図 3.24 WindowsNT のメニュー一覧

### 3.4.6 ホストコンピュータへの接続

TeratermPro が起動されると、図 3.25 が画面に表示されます。そのままリターンキーを押してください。もし TeratermPro が画面に現れなかったら、あなたはソフトウェアの起動に失敗しています。もう一度操作をやり直してください。

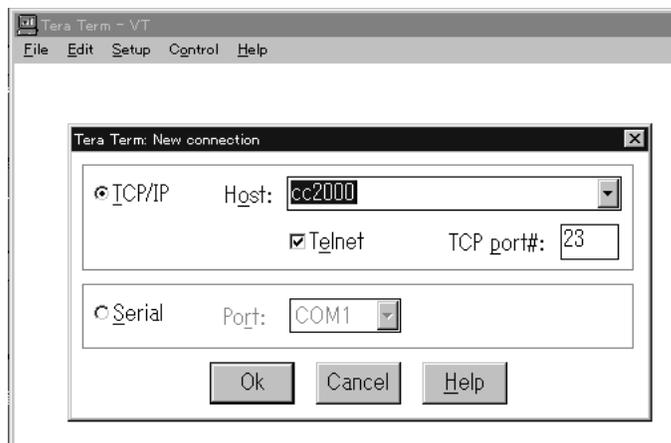


図 3.25 TeratermPRO の初期画面

画面が変わり、login:と表示された状態になります。

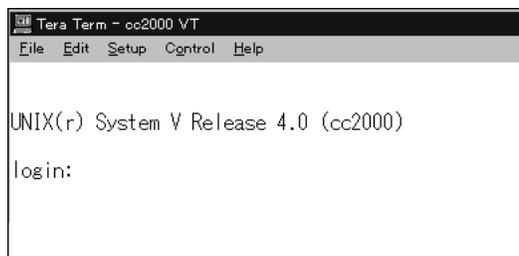


図 3.26 cc2000 のプロンプト

### 3.4.7 さあ、login しよう

いよいよ login です。ここからはもうあなたのキーボードに対する入力はホストコンピュータに対する遠隔操作となっているのです。

ユーザ ID とパスワードを入力しましょう。login:の後にあなたのユーザ ID を入力してください。ユーザ ID には大文字、小文字の区別があるので注意してください。ユーザ ID の入力が完了したら、一度リターンキーを押してください。次にパスワードを入力します。ユーザー名を入力した時とは違い、パスワードは横から見ている他の人にバレないように、タイプしてもその文字が画面に表示されません。キーを打ち間違えないように注意して入力し、もう一度リターンキーを押してください。

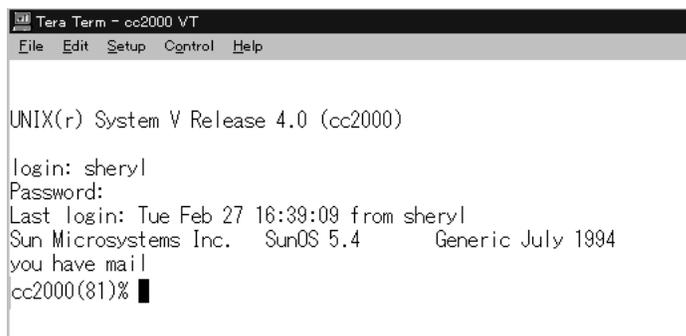


図 3.27 cc2000 のログイン画面

図 3.27 のようになりましたか?もし、ならなかった場合は、3.4.14 の「困ったときの対処」を参照してください。

### 3.4.8 ちょっとメッセージ

login すると、ときには利用者宛の通知、おしらせが以下のように画面に表示されるかもしれません。

```
Sun Microsystems Inc. SunOS 5.3 Generic September 1993
4月15日深夜、ファイルサーバである ccnfs (cc2000) の
レポートが予定されています。
これにともない cc 環境の全てのマシンの利用が出来なくな
ります。夜間に長時間のジョブを実行する予定のある利用者
はそれまでに実行が終るよう注意して下さい。
cc2000(84) %
```

図 3.28 login 直後のメッセージ

よく読んで、自分に関係があるかどうか確認してください。もしも内容に付いてよく判らないことがあれば計算機センターメインカウンターまで連絡するのがいいでしょう。

### 3.4.9 パスワードを変える

もしもこれがあなたにとって Sign 登録後初めての login ならば、ここでまずパスワードを変更しましょう。Sign 登録時のパスワードは機械的に決められたもので、余り安全とは言えません。必ずパスワードをあなた自身が選んだ言葉に変更してください。パスワードを決める際には 19 ページの 2.1.10 で既に述べた事に注意してください。また、コンピュータにはパスワード付け替えの際に安全なパスワードかどうか、簡単なチェックをする機能がついています。メッセージの指示に従ってください。

パスワードを変更するには `yppasswd` コマンド<sup>9</sup> を利用します。上の節で説明したターミナルを見ると、  
cc2000(81)%

などとなっていますね。(但し `cc2000` という部分や括弧内の数字は例と違うかもしれません。) ここで `yppasswd` とタイプし、リターンキーを押します。すると以下のような状態に成りますね。(以下はユーザ ID `ozaki` の例。 `ozaki` の部分にはあなたのユーザ ID が表示されているはずです。)

```
cc2000(81)% yppasswd
yppasswd: Changing password for ozaki
Enter login(NIS) password:
```

指示通りここでは古い、つまり今先ほどまで使っていたパスワードをタイプし、リターンキーを押します。ここでは login の時と同じくパスワードをタイプしている間、横の人の覗き見によってあなたのパスワードがバレないように、タイプした文字は画面に表示されない事に注意してください。表示されなくともタイプした文字はちゃんと入力されていますから、安心して確実に一文字づつタイプして最後にリターンキーをタイプしてください。すると今度は以下のようなプロンプトを表示します。

```
New password:
```

では指示通り新しい、つまり次からこれにしようというパスワードをタイプし、リターンキーを押してください。ここでもタイプした文字は表示されません。今度は以下のようなプロンプトを表示します。

```
Re-enter new password:
```

これはいましがたタイプした新しいパスワードが打ち間違いで無いことを確認するためのものです。もう一度新しいパスワードをタイプしてリターンキーを押してください。今度は以下のような状態になります。

<sup>9</sup> 綴りがちょっと変ですが、間違っはけません。

```
NIS(YP) passwd/attributes changed on ccnic
cc2000(82)%
```

この NIS (YP) passwd/attributes changed が表示されれば成功です。もしもパスワードが短すぎたりタイプミスがあったりしたらなんらかのエラーメッセージが表示されます。途中でおかしくなったと思う場合は C-c (つまりコントロールキーを押しながら C キーを押す) で cc2000(82)% を表示させます。それからもう一度 yppasswd コマンドをやり直します。

よく表示されるエラーメッセージを紹介しておきます。

```
Password must contain at least two alphabetic characters and
at least one numeric or special character.
```

パスワードはアルファベットと数字や記号を混ぜたものにしてください。アルファベットや数字だけのものは推測されやすく、危険です。

```
Password is too short - must be at least 6 characters.
```

パスワードは 6 文字から 8 文字にしてください。(9 文字以上入力することもできますがコンピュータが判別するのは 8 文字目までです。)

```
Passwords must differ by at least 3 positions
```

古いパスワードと変化の少ないものを付けることはできません。

```
Too many failures - try later.
```

失敗しすぎです。もう一度最初からどうぞ。

パーミッションが与えられていません。

何かを失敗したためにユーザ本人と認められませんでした。そのためパーミッション(権限)がないと判断されました。おそらく古いパスワードを間違えて入力したのでしょう。

### 3.4.10 logout をしよう

次は logout です。他にやりたいこともいろいろあるでしょうが、今回は最初の利用でしょうし、まず一通りの操作をやってみるという意味で logout をこの場でやってみることを勧めます。画面上に cc2000(81)% のように表示されていることを確認してください。(但し表示されている文字は若干違うかもしれません。) 以下のように、logout と入力してリターンキーを押してください。

```
cc2000(81)% logout
```

### 3.4.11 端末ソフトを終了する

logout すると自動的に今まで表示していた画面が閉じ、最初の状態(図 3.20)になります。ログインした最初の状態に戻りましたよね。

DigitalPC から telnet 端末環境で cc2000 を利用する場合に必要な手続きの説明は以上です。もちろんこの後パソコンの利用を終了する場合は、正しく決められた手続きで Windows を終了し電源を切ってください。間違ってもこの段階でいきなり電源スイッチを切ったりしてはいけません。

### 3.4.12 ログオフ（シャットダウン）しましょう

画面上に「START」ボタンがあります。画面上のマウスポインター（矢印）をボタンの上に重ね、マウスの左ボタンを押してください。メニューが現れ、そのメニューの一番下の行に「シャットダウン」があります。選択してください。

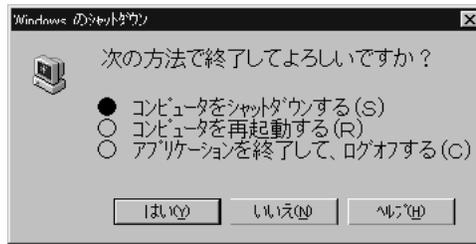


図 3.29 ログオフおよびシャットダウン画面

図 3.29 が画面上に現れます。「コンピュータをシャットダウンする」の左にある丸印をクリックして選択し、リターンキーを押してください。しばらくすると、自動的に電源が切れます。操作の途中で電源を切ったりしないでください。故障の原因になります。

### 3.4.13 さて、さて、

最初の利用はこれでおしまいです。大変よくできました。あとは精進して UNIX ユーザへの道を歩まれるわけですね。それには UNIX での一般的なコマンド操作などについて説明している、4 へと進んでください。

### 3.4.14 困った時の対処

login ができない

- 次のように表示されて図 3.27 にならない場合がある。接続先のホスト名が間違っていて入力されている可能性があります。接続しようとするホスト名をもう一度確認し再度実行してください。



- 次のように login incorrect が表示される。ユーザー名かパスワードが間違っている事が考えられます。大文字、小文字等の入力間違いがないかを確認、再度入力してください。

```
Tera Term - cc2000 VT
File Edit Setup Control Help

UNIX(r) System V Release 4.0 (cc2000)

login: sheryl
Password:
Login incorrect
login: SHERYL
Password:
Login incorrect
login:
```

文字が全角になる

- 次のように、sheryl と入力しているのに「しゑ r y l」と画面上に現れる事がある。これはパソコンの日本語変換機能が起動されている事が考えられる。「Alt」キー（キーボード左下）を押しながら「半角/全角」キー（キーボード左上）を押して、日本語変換機能を止めてください。

```
Tera Term - cc2000 VT
File Edit Setup Control Help

UNIX(r) System V Release 4.0 (cc2000)

login: しゑ r y l
Password:
```

### 3.4.15 マニュアルなど

パソコンのマニュアルは部屋内の本棚に入っています。どうぞ自由に取り出してみてください。但し、持ち帰りはいけません!!。禁帯出です。

次は、71 ページの 4 へ進んでください。

## 3.5 Turbo Linux j3.0 を使う

IBM Netfinity 3000 (以下「Netfinity」)は10号館2,3階の情報処理教室に設置してあります。このコンピュータには2種類のOSがインストールされています。1つはMicrosoft WindowsNT 4.0、もうひとつはTurbo Linux j3.0です。ここではTurbo Linux j3.0を利用します。



図 3.30 NetfinityPC 外観

### 3.5.1 まず部屋へ

10号館の情報処理教室は常時カードロックシステムが働いており、学生証を入り口のドア横の機械に通すことによって鍵が開きます。入室可能な時間帯、曜日、期間に限りがありますので、付録の「情報処理教室の利用について」を一読してください。もしもカードを通して鍵が開かない場合は、カードの向き、表裏、カードを通過させる速度(遅すぎても速すぎても駄目)を変えてやってください。どうしても駄目な場合はカードの磁気情報がおかしくなっている可能性があります。10号館3階の計算機センターメインカウンターまで来てください。

部屋に入るときは、そこで授業をやっていないことを確認しましょう。もし授業中であれば入室できません。

10号館の2,3階の情報処理教室では、授業中の自由利用は認めていません。

部屋に誰も居なかった場合は照明、空調機が切れている場合があります。これらのスイッチは部屋に入ったそのすぐ右手の壁に集中して置いてありますので、自分で操作してください。コンピュータについての質問、要望、部屋の状態について質問、要望があれば計算機センターのメインカウンターまでお願いします。

### 3.5.2 電源を入れる

自分が利用するコンピュータを選びます。これが最初の利用であれば、電源の入っていないマシンを選ぶのがいいでしょう。というのは、もし電源が入っていたら、それは誰かが既に使っているものかも知れないからです。ディスプレイ(テレビの様な画面)は常時電源が入っており、オレンジランプまたはグリーンランプが点灯しているはずですが、



図 3.31 液晶ディスプレイの電源

オレンジランプの時は本体の電源が切れたときの省電力モードを表すオレンジのランプです。グリーンランプは本体の電源が入っており、画面の焼き付け保護のために、黒くなっているだけです。マウスか矢印キーを押すことで、画面が表示されます。

ランプを確認せずに、画面が真っ暗だからと言ってそのマシンの電源が切れているかどうかはわかりません。かならず本体とディスプレイの両方のランプを確認してください。電源が入っていないマシンが見つかったら、その本体上部の電源ボタンを押してください。



図 3.32 電源スイッチ

電源が入ったら、以下の画面が表示されます。



図 3.33 ブートメニュー

キーボード上の矢印「**↑**」、「**↓**」を操作して、選択する OS 名を黒く反転させてください。「Turbo Linux j3.0」が選択できれば、「Enter」キーを押してください。その状態でしばらくお待ち下さい。

### 3.5.3 login する

画面中央には以下のウィンドウが表示されていると思います。



図 3.34 login ウィンドウ

キーボードを使ってまずあなたのユーザ ID をタイプします。ユーザ ID をタイプし終り、リターンキーを一度押せば今度はパスワードを入力できる状態になります。ここでパスワードを入力するのですが、パスワードは横から見ている他の人にバレないように、タイプしてもその文字が画面に表示されません。キーを打ち間違えないように注意して入力し、もう一度リターンキーを押します。うまくユーザ ID とパスワー

ドを入力できれば画面表示が変わります。キーを打ち間違えたりして login に失敗するとウインドウの下の部分に赤い文字で「Login incorrect」と表示されます。この場合はもう一度ユーザ ID のタイプからやり直します。

もしこれがあなたにとって初めての login であればパスワードはユーザ ID 交付時に渡されたものを入れてください。もし何度試しても login できないようであればユーザ ID、パスワードのいずれかが間違っているか、もしもユーザ ID も正しく、また登録作業も完了しているなら、これはパスワードが違っているとした考えられません。計算機センターメインカウンターに連絡してパスワードを強制変更して貰ってください。

### 3.5.4 ちょっとメッセージ

login すると、ときには利用者宛の通知、おしらせが以下のように画面に表示されるかもしれません。

```
#####
# システムのメンテナンスのため、12月25日(金)の 9:00 から 26日(土)の15:00まで
# cc2000 を停止します。ご了承ください。                                     システム管理者
#####
# At Mar. 17th, from 10:30PM to 20:00, cc2000 machine will be stop for system
# maintenance, Please be patient.                                         System Manager
#####
エンターキーを押してください

(エンターキーを押して画面が切り替わらない時はマウスカーソルをこの窓内に移動して
からエンターキーを押してください)
```

図 3.35 login 直後のメッセージ

よく読んで、自分に関係があるかどうか確認してください。確認が済めば、リターンキーを一度押すだけでこの表示を消すことができます。もしも内容に付いてよく判らないことがあれば計算機センターメインカウンターまで連絡するのがいいでしょう。

### 3.5.5 ターミナルはあるかな？

login に成功すると以下のような画面表示になっていると思います。

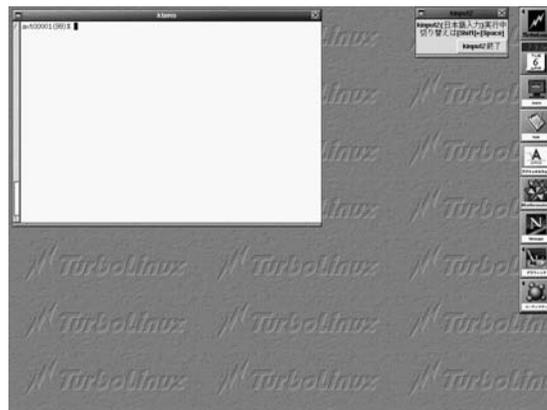


図 3.36 Linux の初期画面

画面の中に以下の様なウインドウがあるかどうかさがしてください。そのようなウインドウが見つければ、ひとつ飛ばして「ターミナルは大丈夫かな？」まで進んでください。もしも見つからなければこのまま次に進んでください。



図 3.37 Linux の kterm

#### ターミナルを起動する

画面上の絵 (  ) を探してください。探し出したらその絵にマウスカーソルを重ね合わせ、マウスの右ボタンを押してください。

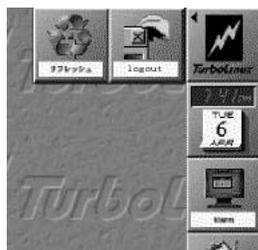


図 3.38 Linux のアイコン

これで KTerm が起動されるはずですが。

ターミナルは大丈夫かな？

ターミナルのウインドウの左隅には `axt00001(11)%` などと表示されていると思います。(但し数字は違うかもしれませんが。)ここで試しに `date` とタイプしてリターンキーを押してみてください。文字をタイプしても `axt00001(11)%` に続いて表示されない場合は、ターミナルウインドウがアクティブになっていません。一度 `axt00001(11)%` 辺りをクリックしてターミナルウインドウをアクティブウインドウにした後、`date` とタイプし直してください。以下のように今日の日付と時間が表示されれば大丈夫、ターミナルを使える状態になっています。

```
%  
axt00001(11)% date  
Thu Apr  8 15:26:46 JST 1999  
axt00001(11)%
```

### 3.5.6 パスワードを変える

もしもこれがあなたにとって Sign 登録後初めての login ならば、ここでまずパスワードを変更しましょう。Sign 登録時のパスワードは機械的に決められたもので、余り安全とは言えません。必ずパスワードをあなた自身が選んだ言葉に変更してください。パスワードを決める際には 19 ページの 2.1.10 で既に述べた事に注意してください。また、コンピュータにはパスワード付け替えの際に安全なパスワードかどうか、簡単なチェックをする機能がついています。メッセージの指示に従ってください。

パスワードを変更するには `yppasswd` コマンド<sup>10</sup> を利用します。上の節で説明したターミナルを見ると、

```
cc2000(81)%
```

などとなっていますね。(但し `cc2000` という部分や括弧内の数字は例と違うかもしれません。)ここで `yppasswd` とタイプし、リターンキーを押します。すると以下のような状態に成りますね。(以下はユーザ ID `ozaki` の例。`ozaki` の部分にはあなたのユーザ ID が表示されているはずです。)

```
cc2000(81)% yppasswd  
yppasswd: Changing password for ozaki  
Enter login(NIS) password:
```

指示通りここでは古い、つまり今先ほどまで使っていたパスワードをタイプし、リターンキーを押します。ここでは login の時と同じくパスワードをタイプしている間、横の人の覗き見によってあなたのパスワードがバレないように、タイプした文字は画面に表示されない事に注意してください。表示されなくともタイプした文字はちゃんと入力されていますから、安心して確実に一文字づつタイプして最後にリターンキーをタイプしてください。すると今度は以下のようなプロンプトを表示します。

```
New password:
```

では指示通り新しい、つまり次からこれにしようというパスワードをタイプし、リターンキーを押してください。ここでもタイプした文字は表示されません。今度は以下のようなプロンプトを表示します。

<sup>10</sup> 綴りがちょっと変ですが、間違っていないです。

Re-enter new password:

これはいましがたタイプした新しいパスワードが打ち間違いで無いことを確認するためのものです。もう一度新しいパスワードをタイプしてリターンキーを押してください。今度は以下のような状態になります。

```
NIS(YP) passwd/attributes changed on ccnic
cc2000(82)%
```

この NIS (YP) passwd/attributes changed が表示されれば成功です。もしもパスワードが短すぎたりタイプミスがあったりしたらなんらかのエラーメッセージが表示されます。途中でおかしくなったと思う場合は C-c (つまりコントロールキーを押しながらCキーを押す) で cc2000(82)% を表示させます。それからもう一度 yppasswd コマンドをやり直します。

よく表示されるエラーメッセージを紹介しておきます。

```
Password must contain at least two alphabetic characters and
at least one numeric or special character.
```

パスワードはアルファベットと数字や記号を混ぜたものにしてください。アルファベットや数字だけのものは推測されやすく、危険です。

```
Password is too short - must be at least 6 characters.
```

パスワードは6文字から8文字にしてください。(9文字以上入力することもできませんがコンピュータが判別するのは8文字目までです。)

```
Passwords must differ by at least 3 positions
```

古いパスワードと変化の少ないものを付けることはできません。

```
Too many failures - try later.
```

失敗しすぎです。もう一度最初からどうぞ。

パーミッションが与えられていません。

何かを失敗したためにユーザ本人と認められませんでした。そのためパーミッション(権限)がないと判断されました。おそらく古いパスワードを間違えて入力したのでしょう。

### 3.5.7 logout する (セッションを終了する)

他にやりたいことも色々あるでしょうが今回は最初の利用でしょうし、まず一通りの操作をやってみるという意味で、logout をこの場でやってみることを勧めます。

画面上部に  ボタンがあります。それを押すと、「logout」の絵が現れます。絵に画面上のマウスポインターを重ね合わせ、マウスの左ボタンを押してください。

しばらくすると最初の画面(図 3.34)に戻ります。画面上の「電源 OFF」ボタンにマウスポインターを重ね合わせ、マウスの左ボタンを押してください。しばらくすると、電源が切れます。操作の途中で電源を切ったり、叩いたりしないでください。故障の原因になります。



図 3.39 ログオフ画面

これが Linux での logout 操作です。実際には logout という名前のコマンドも操作も実行しないことに注意してください。Linux ではつまり logout とはセッションの終了だということです。セッション終了の操作を行なって数十秒すると、最初に電源を入れて login 操作をする直前の状態に戻って図 3.34 が表示されます。もしもその状態にならない、つまり正しくセッションが終了できなければ、計算機センターメインカウンターまで連絡ください。そのまま放置するのはいけません。また電源を無理に切るのはコンピュータが壊れるので厳禁です。

### 3.5.8 シャットダウンする

セッションが終了できれば、最初に電源を入れて login する直前の状態に戻ります。画面上の右下に「電源 OFF」ボタンがあります。画面上のマウスポインターを重ね合わせ、マウスの左ボタンを押してください。

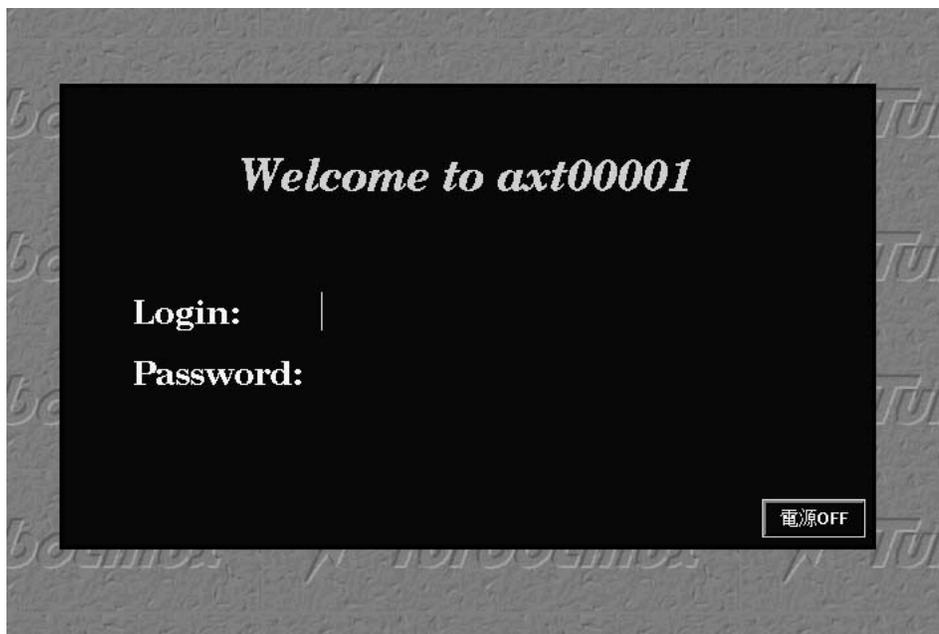


図 3.40 電源 OFF ボタン

すると数十秒で画面が白くなって、電源が自動的に切れます。この操作をシャットダウンと呼んでいます。

電源が自動的に切れない場合は何かトラブルがあったと思われます。決して電源スイッチを切らないでください。また、おかしなまま放置するのもいけません。計算機センターメインカウンターまで連絡ください。

### 3.5.9 部屋を出る

マシンの電源を切って部屋を出るとき、他に使用者がいなければ照明も切って行きましょう。

### 3.5.10 さて、さて、

最初の利用はこれでおしまいです。大変よくできました。あとは精進して UNIX ユーザへの道を歩まれるわけですね。それには UNIX での一般的なコマンド操作などについて説明している、4 へと進んでください。

次は、71 ページの 4 へと進んでください。

## 第4章 UNIXそれから

ここでは計算機センターが管理している UNIX 環境を例に取りながら、比較的一般的な UNIX 環境の利用方法を説明します。ここでの説明は網羅的なものではなく、部分を取り上げて曖昧に説明しています。これは本文の読者のコンピュータそのものについての知識のハードルを高くしすぎないためです。読者が UNIX、つまりコンピュータの利用に慣れて行くにしたがって自分でマニュアル、書籍、ネットワークなどから情報を常に取り込んで理解を深めていくことが大切です。

### 4.1 基礎知識をもう一度

#### 4.1.1 login

UNIX コンピュータを利用する為にはまず初めに、誰がこれから利用するのかをコンピュータに伝える login<sup>1</sup> と呼ばれる作業をしなければなりません。本文では既に login が済んでいるものとして説明を続けます。また、ターミナルが起動され、コマンドを入力できる状態になっていることを前提にしています。機種ごとの login の方法、ターミナルの起動の方法については 2 の「UNIX はいかが？」をご覧ください。

ところで login するのは良いのですが、ときどき logout<sup>2</sup> しない(もしくはセッションを終了しない)人がいます。logout しないと他の人がそのコンピュータを使えないばかりか、逆に他人に悪用されてしまいますから、しばらくしてまた戻ってくる場合でも席を外す際は logout するべきです。

#### 4.1.2 キー表記

キーボードから様々な文字列を打ち込むことを「タイピング」と呼んでいます。「abc」と打ち込むことを「abc とタイプする」「abc と入力する」などと表現することもあります。UNIX では通常のアルファベットや数字以外に、様々なキーがあります。例えばアルファベットの A と書かれたキーを押せば、小文字の「a」が入力されるでしょう。もしも大文字の「A」をタイプしたければ「シフトキー」を押しながら A キーを押すことになります。これらのことは既に 2 の 2.3 で説明しました。ですからこれ以降は「aBc」とタイプする、と書けば「アルファベットの A キーを押して、シフトキーを押しながら B キーを押して、C キーを押す」というように解釈してください。同様に、以下のような記述で様々なタイピングを表現します。

---

<sup>1</sup> ろぐいんと読むのでしたね。

<sup>2</sup> ろぐあうと読むのでしたね。

重要な語句の説明です。覚えてください!!

<Return>	リターンキーを意味します。
<Delete>	削除キーを意味します。一般的には Delete キーに当たります。
<Space>	空白キーを意味します。
<Tab>	タブキーを意味します。一般的にはアルファベットの Q の左にあります。
<ESC>	エスケープキーを意味します。一般的には数字の 1 キーの左にあります。
<Control> X	コントロールキーを押しながら X キーを押すことを意味します。
<ESC> X	エスケープキーを押してから X キーを押すことを意味します。
C-X	コントロールキーを押しながら X キーを押すことを意味します。
M-X	メタキー（一般的には <ESC>キーに同じ）を押してから X キーを押すことを意味します。

### 4.1.3 カーソル

タイピングしている最中に、次にキーを押したらどこにその文字が入力されて表示されるかを判り易くするために表示する目印のことを「カーソル (cursor)」と呼んでいます。形は状況に応じて様々で、丁度一文字分の大きさの（黒い長方形）だったり、|（縦棒）だったりします。他の文字と見分けが付き易いように点滅している場合もあります。

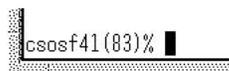


図 4.1 カーソルの例（黒い長方形）

## 4.2 コマンド

### 4.2.1 コマンドって何だ？

コンピュータというのは機能の集合体だと考える事ができます。これからあなたが利用しようと思っているコンピュータの中には星の数ほどの（いやそんなに無いな、山ほどの）機能があり、その中から利用者は自分の希望の機能を選んで実行させるのです。山ほどの機能にはそれぞれ固有の名前が付けられており、利用者は機能をその名前を選ぶというわけです。逆に言うと機能の名前を知らなければどんなに便利な機能がコンピュータの中にあっても利用できませんし、名前を間違えて指定すれば、望みの機能とは違う機能が働き出します。この「機能にそれぞれ付けられている固有の名前」をコマンドと呼んでいます。山ほどあるコマンドを利用者が覚え易いように、その機能を連想し易い名前がコマンド名として付けられています。

あるコマンドを入力するとは、そのコマンドに対応する機能を実行するように指示するという事です。

### 4.2.2 プロンプト

コンピュータの利用というのは実際にはこのコマンド実行の繰返しだと言えます。コンピュータが「コマンドをどうぞ」というメッセージを表示し、利用者がコマンドを入力する。コンピュータはコマンドの結果をメッセージとして表示し、「次のコマンドをどうぞ」と、またメッセージを表示する。そして利用者が再びコマンドを入力する、という具合です。見方を変えればコンピュータと利用者はメッセージとコマンドで「対話」しているようにも見えます。このコンピュータが利用者との対話のために用いている「次のコマンドは何ですか？」という催促メッセージのことを一般に「プロンプト (prompt)」と呼んでいます。

cc 環境での一般的なプロンプトは `cc2000(81)%` などと表示されます<sup>3</sup>。cc2000 で login した直後に見える ウィンドウの左隅に見つけることができるでしょう。

### 4.2.3 簡単なコマンド

#### date コマンド

ここで一つ非常に簡単なコマンドを実行してみましょう。date コマンドです。プロンプトに続いて `date` <Return>とタイプしてください。以下のように時間が表示されるでしょう。つまり date は日付と時間を表示するという機能のコマンドです。

以下に示す例は実際にあなたが試してみたときとは結果が違うでしょう。この例を試したときと、今とでは状況が違っているからです。これ以降に挙げる例も、全く同じ結果が表示されるとは限らないことに注意してください。

```
cc2000(81)% date
1997年02月25日(火)15時14分37秒 JST
cc2000(82)%
```

もしも `date` とタイプしたのにそのとおりに入力できない様な場合は 2 の 2.4.5 をチェックしてください。

もしも以下のように `Command not found` などというメッセージが表示されてしまった場合はコマンドの打ち間違いです。もう一度、今度は正確にコマンドをタイプしてください。

```
cc2000(81)% data
```

---

<sup>3</sup> 括弧の中の数字の意味は後でわかります。例と違って気にしないでください。

```
data: Command not found
cc2000(82)%
```

この正しくない操作を利用者が実行しようとしたためにコンピュータの方から「それは駄目だったよ」という意味のメッセージが返ってくる時があります。このようなメッセージを一般に「エラーメッセージ」と呼んでいます。

## hostname コマンド

hostname コマンドで自分が使っているコンピュータのホスト名を得られます。

```
cc2000(82)% hostname
cc2000
cc2000(83)%
```

上記の例から分かるように、cc 環境ではホスト名はプロンプトに含まれていつでも表示されています。

## 4.2.4 引数とオプションのあるコマンド

### finger コマンド

finger コマンドで、利用者の情報を得ることができます<sup>4</sup>。finger に続いて調べたいキーワードをタイプして<Return>です。ここでは sigeta さんについての情報を表示させてみましょう。

```
cc2000(127)% finger sigeta
Login name: getter14                In real life: Sigeta Sintarou
Directory: /NF/home/g930/getter14  Shell: /usr/local/bin/tcsh
Last login Wed Feb 28 15:16 on pts/69 from e51nf21
No unread mail
No Plan.

Login name: sheryl                  In real life: Sigeta Hiroyuki
Directory: /NF/home/syokuin0/sheryl Shell: /usr/local/bin/tcsh
On since Mar 13 15:47:12 on pts/39 from c1nf01
37 minutes Idle Time
No unread mail
No Plan.
cc2000(128)%
```

例から判るように二件の情報が表示されました。両方ともユーザ ID ではなくフルネームの部分に指定したキーワードがマッチしたから表示されたのですね。

finger ではコマンドに続いてキーワードを指定しましたね。この様にコマンドの後ろにそのコマンドが実行するために必要な情報を付ける場合があります。このコマンドの後ろに付けるものを「引数 (ひきすう)」と呼んでいます。パラメータなどと呼ぶ場合もあります。引数は常にコマンドの後ろに書きます。逆に言うとコマンドが必ず一番前に来ます。

---

<sup>4</sup> finger の結果は間違ふことがあるのであまりあてにしないでください。

また、コマンドの処理内容を少し変える指示を与えることができる場合もあります。例えば `finger` コマンドの表示は少し長いので、これを短くするために `-s` という指示をコマンドの後ろ、キーワードの前に与えます。

```
cc2000(128)% finger -s sigeta
Login      Name          TTY          Idle    When      Where
getter14  Sigeta Sintarou pts/69       <Feb 28 15:16> e51nf21
sheryl    Sigeta Hiroyuki pts/39       41 Wed 15:47  c1nf01
cc2000(129)%
```

では、ユーザ ID `sheryl` さんの情報だけが見たいと思った時は、`finger` コマンドに `-m` という指示を加えます。

```
cc2000(131)% finger -m sheryl
Login name: sheryl          In real life: Sigeta Hiroyuki
Directory: /NF/home/syokuin0/sheryl  Shell: /usr/local/bin/tcsh
On since Mar 13 15:47:12 on pts/39 from c1nf01
45 minutes Idle Time
No unread mail
No Plan.
```

ユーザ ID `sheryl` さんの情報が表示されましたね。

この `-s` や `-m` などのように、コマンドの処理内容を少し変化させるような指示を「オプション」もしくは「コマンドオプション」と呼んでいます。オプションは常にコマンドの後ろに書きます。逆に言うとコマンドが必ず一番前に来て、それからオプション、その後その他の引数という並びになります。オプションは大抵 `-` 記号（引き算記号、ハイフン）に導かれています。また、オプションが複数並ぶ場合については `-s -m` と並べて書くこともできますし、縮めて `-sm` と書くこともできる場合があります。つまり上記の例だと `finger -sm sheryl` と書いても同じ意味です。

#### 4.2.5 対話的なコマンドとそのサブコマンド

##### bc コマンド

`bc` コマンドで、簡単な四則演算電卓を利用することができます。`bc <Return>` でカーソルが画面の左端で止まります。そこで四則演算の式を一行入力して `<Return>` すれば、すぐ次の行に結果が表示されます。プロンプトは特に表示されませんが、そこでまた式を入力すればまたその結果が表示されますから計算は何度でも繰り返して行えます。

```
cc2000(65)% bc
1 + 2 * ( 3 + 4 )
15
```

この状態では、コンピュータは四則演算の式を計算する `bc` コマンドの最中で、それ以外の利用者の指示は受け付けられません。式以外、例えば `date` のような通常のコマンドをこの状態でタイプしても以下のようにエラーメッセージを表示するだけです。

```
date
syntax error on line 1 (null)
```

計算を止めてコマンドプロンプトの状態に戻りたければ quit <Return>です。

```
quit
cc2000(66)%
```

bc などのように、コマンドの中には実行すると更にユーザからの指示を待つような（つまり「対話的な」）動きをするものもあります。bc における quit などのようにコマンドの処理中にユーザがそのコマンドに対して更に細かい指示を与えるコマンドの事を「サブコマンド」と呼んだりします。

#### 4.2.6 コマンドの使い方を調べる

さて、コマンドと一口に言っても、どうやらその機能と名前を覚えるだけでは済みそうにないことが判りますね。コマンドをどれか一つ取ってみても、それぞれ独自のオプション、引数、もしくはサブコマンドなどがあり、どのコマンドにどんなオプション等があったかを覚えないと実際困りそうです。でもそんなものいつでも全部覚えておけるほどみんな暇ではありません。その為にコマンドの正確な機能、使い方を教えてくれるコマンドがあります。man<sup>5</sup> コマンドです。例えば先ほど取り上げた finger コマンドのオプションなどを忘れてしまった場合にはすかさず man finger <Return>です。

```
csosf01(79)% man finger
finger(1)                                     finger(1)
NAME
  finger, f - Displays user information
SYNOPSIS
  finger [-bfhilmpqsw] [user ...]
```

The finger command displays information about the users in the passwd file.

というわけです。このマニュアル表示は大抵一画面では収まり切りませんから、一ページごとに一旦停止します。<Space>で一ページ分送ります。<Return>で一行分送ります。この表示が送られて行く様子を「スクロール」と呼んでいます。全てを表示し切ったら man コマンドは終了します。一旦停止している状態で、q キーを押すと最後まで表示せずにその状態で man コマンドの表示を終了させることができます。h キーか、?キーのいずれかを押せば、どのようなキーを押せば一旦停止後にどのような操作が効くかを表示してくれます<sup>6</sup>。

#### man によるマニュアルの書式

man コマンドが表示する内容をよく見てみると、幾つかの項目に分けて説明してあるのが判るでしょう。finger の場合だと NAME, SYNOPSIS, DESCRIPTION, FLAGS, EXAMPLES, FILES, RELATED INFORMATION という具合です。上記の項目分けは、全ての UNIX において全く同じではなく、幾らか違う部分もありますが大抵は共通の形式で書かれています。以下にその項目の意味を書いておきます。

- NAME, 名前  
コマンドの名前とその概要。

---

<sup>5</sup> 「マニュアル (manual)」 のつもりで man です。

<sup>6</sup> つまりこれらは「表示一旦停止機能」のサブコマンドというわけです。

- **SYNOPSIS, 形式**

コマンドの形式。コマンドに適用できるオプション、引数などを列挙します。

ここでの表記にはルールがあって、例えば `finger` の例を以下に挙げると、

```
finger [-bfhilmpqsw] [user ...]
```

この [ ] に囲まれた部分は「なくてもいいよ」つまり省略可能だという事を意味しています。特に上記の例のように [ ] に多くのオプション文字がくくられていた場合は、その中のどれでもピックアップして同時に与えていいよ、という意味です。また、... は、その直前のものを繰り返して書いてもいいよという意味です。つまり `finger` ではキーワードを複数書けるのですね。コマンドにオプションや引数をつけてタイプする場合、ここに表示された順番に注意してください。

- **FLAG, OPTION, フラグ, オプション**

それぞれのオプションの働きについて詳しく書いてあります。あるオプションとは相反する指示だから、これとこのオプションは同時に指定してはいけないよ、などということも書いてあります<sup>7</sup>。

- **DESCRIPTION, 機能説明**

コマンドの詳細説明。コマンドの機能が詳しく書かれています。ここにオプションの説明を含めている UNIX もあります。

- **FILES, ファイル**

コマンドに関係するファイル<sup>8</sup> の名前が列挙されます。

- **SEE ALSO, RELATED INFORMATION, 関連項目**

コマンドに関連する項目。深い関係のあるコマンドなどが列挙されます。この項目は再び `man` コマンドで参照できますから、このコマンドのマニュアルだけを読んでよく判らない場合はここを追い掛けて行くのがお勧めです。

- **BUGS, バグ**

コマンド使用上の制限事項。コマンドの動きがどうもおかしいという時は注意してみましよう。

## コマンド名を調べる

コマンドの名前が判らないのだけれど、このような機能を持ったコマンドを探したい、という時にも `man` コマンドは有効です。キーワードでコマンドを検索するオプションとして `-k` オプションがあります。 `man -k password` などとすれば `password` に関係のあるコマンドなどの一覧が表示されます。結構沢山出ますが一行で一つのコマンドを紹介してくれています。一番目の項目がコマンド名、その次の括弧に囲まれた数字がマニュアルの分類番号で、残りがコメントです。

```
cc2000(86)% man -k password
```

```
conflict (8)          - search for alias/password conflicts
lock (1)              - Requests and verifies a user password
passwd (4)            - Password files
passwd, chfn, chsh (1) - Changes password file information
popwrd (8)            - Sets password for a POP subscriber
printpw (8)           - Outputs the contents of the password database
pwck, grpck (8)       - Checks the password and group files for inconsistencies
```

---

<sup>7</sup> こういうのを排他的なオプションなどと表現することもあります。

<sup>8</sup> 後述。4.4 参照

```

yppasswd (1)          - change password in Network Information Service (NIS)
yppasswdd, rpc.yppasswdd (8) - server daemon for modifying the Network Information Service (NIS) password file
cc2000(87)%

```

このような感じですね。上記の括弧に囲まれたマニュアルの分類番号の一覧を以下に示しておきます。この分類のことをセクションと呼んでいます。

セクション番号	分類
1	ユーザコマンド（一般利用者の為のコマンド）
2	システムコール（プログラム言語から利用します）
3	関数（プログラム言語から利用します）
4,5	各種ファイルフォーマット
6	ゲームとデモ
8	保守用コマンド（システム管理者が利用します）

ところでこのセクション番号の割当てですが、UNIX によってちょっと違いがあります。セクション 1, 2, 3 位まではどの機種でも同じなのですが、4, 5 あたりについては上記の表は余り当てにならないことに注意してください。

さて、例に挙げた password キーワードでのマニュアル検索ではセクション 1 とセクション 4 と両方に passwd という項目がありましたね。ここで man passwd とすると常に前の方だけ、つまりセクション 1 の方についてだけが表示されます。このような状況でセクション 4 の passwd について知りたい場合は、man コマンドでセクション番号を明示してやります。ここでちょっと気にしなければいけないのはセクション番号の指定の仕方が UNIX によってまちまちだということです。現在の cc 環境には 2 種類の指定方法があります。以下に Enterprise3000(Solaris2) の場合と Linux の場合の指定方法をそれぞれ示します。世の中には他にも様々な種類の UNIX がありますが、大抵どちらかです。

機種	OS 名称	コマンド記述
Enterprise3000	Solaris2	man -s 4 passwd
Netfinity3000	Linux	man 4 passwd

#### 4.2.7 UNIX によるコマンドの違い

UNIX は一種類ではなく各メーカーからたくさんの種類の UNIX コンピュータが出荷されています。UNIX はそれぞれのメーカーで独自に改造され、コマンドも少しづつ動きが違います。違う UNIX を採用している場合には、コマンド名は同じでもオプションが違うという事はよくあることです。先述の man コマンドもその例です。

cc 環境は ULTRA1(Solaris2), Enterprise3000(Solaris2), Netfinity3000(Linux) の UNIX マシンの混成です。このドキュメントは一般的な UNIX なら大抵あてはまるようにして書いていますが、今後例示されるコマンドのうちのいくらかはそのとおりでは動作しない可能性があります。おかしいな、と思ったらすぐ man コマンドで確認する習慣を身につけましょう。

#### 4.2.8 トラブルからの脱出

UNIX を操作していて、どうにもおかしな状態になってしまって困ることがあります。もう一度始めからやり直したいんだけど、今どういう状況なのかよく判らないなあどうしようどうしよう、こんなとき...

C-c などの表記に見覚えの無い人は 4.1.2 をもう一度確認してください。

とにかくコマンドを中断して最初からやり直したい

C-c を試してください。何度か C-c するとうまくプロンプトに戻る場合があります。

でもキー入力が全然受け付けられていないようなんですけど

C-q を試してください。C-s でキー入力をロックしてしまう場合が時々あります。C-s の解除が C-q です。C-q の後でならキー入力が効くのでは無いでしょうか。

正確な表現をすると C-s でロックされているのはキー入力ではなく、画面表示です。つまりキー入力は受け付けられているのだけれども、その結果の表示がロックされているので、あたかもキー入力が効いていないかのように見えるのです。注意して見れば C-q の後に今までタイプしていた分と、その結果表示が一気に画面に表示されるはずですよ。

それでも C-c が効いてないようなんです

ええい、仕方がありません。C-z を試してください。それで Suspended などと表示されてシェルのプロンプトが表示されたらしめたものです。その状態ですぐさま `kill %` とやってください。これをしないとプロセス<sup>9</sup>が残ってしまいます。そのあとは普通にコマンドが打ち込める状態になっているのではないのでしょうか。

画面表示がどうにもおかしくなっているんです

Mule 利用中であれば C-1 を試してください。<Control>と英字の L です。それ以外のコマンドなどで画面が乱れている場合はとりあえず C-c などしてシェルのプロンプトまで戻り、そこで `tset` コマンドを試してください。それでも駄目なら `cls` コマンド<sup>10</sup>を試してください。

それで戻らなければ `login` し直すのが早いですね。

でもやっぱりどうにもならないんです

仕方がありません。計算機センターメインカウンターまで連絡ください。情報処理教室なら待機中の補助員に聞くか、備え付けの電話をお使いください。図書館にも補助員が待機しています。聞くは一時の恥。一度聞いておけば次からは自分で対応できますし、自分の能力アップになります。

どんな状況におちいてもいきなりコンピュータの電源を切ってはいけません。コンピュータは正しい方法で終了させないと故障してしまいます。補助員にも連絡がつかないときはキーボードの上にメモを残して「動かなくなってしまいました。後の対応をお願いします。」などを書いて電源をつけたままにしておいてください。

---

<sup>9</sup> 動作中のプログラムのことです。

<sup>10</sup> `cc` 環境独自のコマンドです。一般の UNIX にはありません。

## 4.3 シェル

今までコマンドの例を挙げてきました。その時、コンピュータそれ自体が利用者からのコマンドを受け付けて逐一実行してくれているように書いてきましたが、じつはそれは正確な表現ではありません。いままで「cc2000(81)%」などのプロンプトを表示して、利用者からのコマンドを受け付けてくれていたのは「シェル」と呼ばれるプログラムだったのです。

コンピュータはいろいろなプログラムを実行できます。逆に言うと、コンピュータが実行できるのはプログラムだけです。利用者のキーボード入力からコマンドを受け付けてそれに対応するプログラム<sup>11</sup>を実行するための仕掛けがシェルなのです。勿論シェルもコンピュータの中ではプログラムで実現されていますから、シェルプログラムなどとも呼ばれます。とにかく利用者がキーボードからのコマンドでコンピュータを操作するときに、利用者とコンピュータを仲介してくれるプログラムなのだと考えてください。

### 参考

シェルは一種類ではなく、何種類もあります。一般的に UNIX でシェルと言えば `csh`<sup>12</sup> がポピュラーなのですが、`cc` 環境では `tcsh`<sup>13</sup> と呼ばれるシェルを標準的に採用しています。`tcsh` は `csh` の拡張版のようなもので、基本的な動作は `csh` と同じだと考えてください。世間一般で売られている `csh` のための書籍に書いてある事項は殆どそのまま `tcsh` にも適用できますので、安心して `csh` の参考書を使って貰って結構です。

### 4.3.1 コマンド入力時の編集

シェルには便利な機能があって、コマンドの入力時に利用者のタイピングを手助けしてくれたりします。今までタイピングで間違えたときには、`<Delete>` で一文字づつ戻って間違えたところから打ち直してくれと書いてきました。確かにこれが確実な方法なのですが、しかし例えば `fonger -sm sheryl` とタイプし終わったところで「あっ、`fonger` が `fonger` になってる！」と気が付いた場合、最初から全て打ち直しになって悲しい思いをすることになります<sup>14</sup>。

しかし安心してください。ここでカーソルキーを紹介しましょう。カーソルキーとはリターンキーの少し右辺りに配置してある矢印キーのことです。カーソルについては既に 4.1.3 で説明しましたが、このカーソルが左矢印 ( `<` ) を一つ押すことによって一つ左に移動します。

上記の悲しい `fonger` の例だと、左矢印キーを十数回押して `n` までたどり着き、そこで `<Delete>` を押して `o` を消し、それから `i` を押します。これでめでたく `fonger -sm sheryl` が完成しましたね。ここで元氣よく `<Return>` とすれば完成したコマンドで実行が行なわれます。

カーソルは右にも動きます。右矢印 ( `>` ) キーを押すことでカーソルが一つ右に移動します。そして同じように文字を消したり挿入したりできます。これで自由自在にカーソルを左右に移動して編集できますね。

カーソルを左右に動かしたりする編集キーには他にも幾つかあります。以下に一覧を載せておきます。

---

<sup>11</sup> つまりそれがコマンドの本体なのです。

<sup>12</sup> 「しーしえる」と読んでください。

<sup>13</sup> 「ていーしーしえる」と読んでください。

<sup>14</sup> 特にこのような打ち間違いをするのは初心者の方が多く、タイピングに慣れていない初心者にとっては非常に悲しいものです。

キー	アクション
左矢印 ( )	一文字分左へ
C-b	左矢印に同じ
右矢印 ( )	一文字分右へ
C-f	右矢印に同じ
C-a	コマンド行先頭 (左端) へ
C-e	コマンド行末尾 (右端) へ
<Delete>	カーソル位置の直前の一文字を消去
C-k	カーソル位置から末尾までを消去
C-u	コマンド行全てを消去

これでもう悲しい思いをせずに済みますね。

### 4.3.2 ヒストリ

コマンドを何度もタイプしていると、非常によく似たコマンドを何度も繰り返したり場合によっては何度も同じコマンドを繰り返している事に気が付くでしょう。幾つか前に打ち込んだあのコマンドをもう一度!と思うこともあるでしょう。もっと悲しい場面としては非常に苦労してタイプした長いコマンドが、実はタイプミスを含んでいてもう一度全部タイプし直さなくてはならない、という場合です。

しかし安心してください。そんな時に便利なのがヒストリ ( history ) 機能です。一つ前のコマンドを呼び戻すにはカーソルキーの上矢印 ( ) キーを一度押します。そこで現れた一つ前のコマンドも、左矢印キーや右矢印キーでカーソルを移動させながら編集し直して実行することができます。二つ前のコマンドを呼び戻すには上矢印キーをもう一度 (つまり二度) 押します。上矢印を押しすぎて、目標のコマンドより戻りすぎた場合は下矢印を押します。

キー	アクション
上矢印 ( )	一つ前のコマンド
C-p	上矢印に同じ
下矢印 ( )	一つ次のコマンド
C-n	下矢印に同じ

ここで history コマンドを試してください。一体どれだけのコマンドを覚えているかが判ると思います。上下の矢印キーによって、このヒストリの中を上下することができるというわけです。

```
cc2000(86)% history
  7 21:20 goto label
  8 21:20 man ls
..... (中略)
 82 21:48 finger sheryl
 83 21:48 date
 84 21:48 finger -m sheryl
 85 21:48 finger -sm sheryl
 86 21:48 history
cc2000(87)%
```

これでまた一つ悲しい思いをせずに済むようになりましたね。

### 4.3.3 イベント

また、history のリストを見ると、左に番号があるのが判ります。この番号はプロンプトの括弧の中に出てくる数字に一致しています。この番号で、何番目をもう一度実行せよ、という指示もできます。例えば 84 番目のコマンドを（もしくは 84 番目のイベントを）もう一度実行したいというときは !84 です。

```
cc2000(88)% !84
finger -m sheryl
Login name: sheryl                In real life: Sigeta Hiroyuki
Directory: /NF/home/syokuin0/sheryl  Shell: /usr/local/bin/tcsh
On since Mar 13 15:47:12 on pts/39 from c1nf01
45 minutes Idle Time
No unread mail
No Plan.
cc2000(88)%
```

84 番目のコマンド `finger -m sheryl` が念のために !84 のすぐ次の行に表示されているのが判りますね。

また番号ではなく、一番最近に実行した `x` で始まるコマンドをもう一度実行するという指示も可能です。上記の例の状態、`!d` とすると 83 番目の `date` にマッチして、それが実行されます。

```
cc2000(89)% !d
date
1997年02月25日(火)18時23分31秒JST
cc2000(90)%
```

83 番目のコマンド `date` が念のために !d のすぐ次の行に表示されているのが判りますね。コマンド名のマッチングの為に ! に続けて指定する文字は何文字でも構いません。書いた文字数の分だけでマッチングし、一番最近のコマンドから逆にさかのぼって一番最初にヒットした（適合した）コマンドが実行されます。もしもさかのぼってヒットするコマンドがなかった場合は「そんなイベントはない」という意味のエラーメッセージが表示されます。実際「イベント」という、この機能にまつわる名前を見るのはこのときぐらいです。

これで随分幸せになりましたね。（^\_^）<sup>15</sup>

---

<sup>15</sup> これはフェイスマークといって、文字を組み合わせで顔を表しています。ほーら、笑っている顔に見えますね?:-)（このマークは本を横にして見てください）

## 4.4 ファイル

ファイルとは何かということの本質的に理解できるようになるにはコンピュータの構造を知るところから始めなくてはなりません。これはファイルがコンピュータが発展する過程での歴史的経緯から発生し、現在に至っているからです<sup>16</sup>。

ここではファイルとは何か、という事については余り言及せず、その仕掛けを利用するとこんな事ができる、というところを説明します。

### 4.4.1 でもやっぱりファイルって何？

気になる人の為に少し中途半端なものになるのを覚悟で「ファイル」とは何か説明しましょう。気にならない人は読み飛ばしてください。

身のまわりの電化製品を見渡すと、ディスクやテープなどの「記録媒体」などと呼ばれるものを容易に見つけることができると思います。例えば音楽を聞くのに使っているCD（コンパクトディスク）や、ビデオテープなどです。前者はキラキラ光る面に溝を刻んで、後者は茶色の鉄粉が塗ってあるテープに磁石で印を付けながら、とにかく「なにか」を記録していきます。UNIX コンピュータにもこれと同じように「なにか」を記録できる「ディスク」が幾つも付いています。きっとあなたも今までにワープロ用のフロッピーや、音楽用の MiniDisk など、コンピュータ用のディスクの類似品を目にしたことがあると思います。このディスクの中にいろんなものを記録していくわけですが、コンピュータのディスクはCDみたいに交換できない上に、UNIX コンピュータは大勢の人が使っているので、整理して「もの」を配置しないとどこに何を記録したのか判らなくなってしまう<sup>17</sup>。CDだってアルバム一枚に幾つもの曲を入れてますね。そうしないと5曲目だけ取り出すなんていう事ができなくて不便です。

コンピュータのディスクの中にもものを記録する場合、記録はそれぞれ他の記録と混同しないように分割して残され、他の記録と区別するためにそれぞれ名前が付けられています。このそれぞれ名前を付けられたひとかたまりの情報の集まりを「ファイル」と呼んでいます。つまりコンピュータのディスクの中にはそれぞれ名前を付けられたファイルが幾つも記録されているということです。

### 4.4.2 ファイルの一覧を見る

UNIXに限らず現在市場に出ている殆どのコンピュータはファイルという形で情報（データなど）を格納しています。丁度様々な記録を紙に書いて、ファイル（file、書類差し）に入れて保存するようなものです。あなたの机に文書整理用のファイルがいくつも置いてあるように<sup>18</sup>、コンピュータの中のあなたの記録場所にも幾つもファイルを置くことができます。幾つも作ることができるので、それぞれを区別するために名前が付けられています。「ファイル名」などと表現します。

ファイル名には実際には殆ど全ての文字が利用できます。しかし様々な理由から、ファイル名として利用する文字としては、アルファベット大文字、小文字、数字、\_（アンダースコア）、-（ハイフン）、.（ピリオド）程度にするべきです。これら以外の記号文字、漢字などはファイル名としては利用しない方が無難です。また、ファイル名の長さは255文字が最大です。

ls<sup>19</sup> コマンドで、今あるファイルの名前の一覧をチェックすることができます。

<sup>16</sup> などと書いていますが、実はこの文章を書いている人がよく理解していないから説明できないのです。すみません。

<sup>17</sup> 長らく使っている留守録用のビデオテープのように

<sup>18</sup> ない人も、まああると思ってください。

<sup>19</sup> 字が判りにくいかもしれませんが、英小文字の L と S です。list を縮めたつもりなのです

```
cc2000(82)% ls
Apps      Mail      Wnn6
cc2000(83)%
```

つまり「Apps」「Mail」「Wnn6」という名前の3つのファイル<sup>20</sup>があるということです。

#### 4.4.3 試しにファイルを作ってみましょう

ファイルの一つの利用方法として、あなたが実行したコマンドの記録を取る方法を紹介します<sup>21</sup>。script ファイル名 <Return>で、ファイル名のファイルにコマンドの実行結果が記録できます。ここでは例として時刻、実行しているマシンのUNIXの種類、ある月のカレンダーを表示させるコマンドを実行した記録を取ります。ファイル名として、とりあえず test を指定してみます。script コマンドの終了は exit です。つまり今回のサンプルでのタイピングは、

```
script test <Return> date <Return> uname <Return> cal 7 1999 <Return> exit <Return>
となります。
```

```
cc2000(83)% script test
Script started, file is test
cc2000(81)% date
1997年03月5日(水)23時22分51秒
cc2000(82)% uname
SunOS
cc2000(83)% cal 7 1999
      7月1999
日 月 火 水 木 金 土
          1  2  3
 4  5  6  7  8  9 10
11 12 13 14 15 16 17
18 19 20 21 22 23 24
25 26 27 28 29 30 31
cc2000(84)% exit
cc2000(85)% Script done, file is test
```

test という名前のファイルができた事を ls コマンドで確認しましょう。

```
cc2000(84)% ls
Apps      Mail      Wnn6      test
cc2000(84)%
```

#### 4.4.4 ファイルの内容を見る

cat ファイル名 <Return>で、できたファイルの内容を表示する事ができます。

```
cc2000(85)% cat test
```

<sup>20</sup> cc 環境では作った覚えがなくても、上記の4つのファイル程度は既にホームディレクトリ(後述)に作られています。

<sup>21</sup> 学生のレポート提出などに有効でしょう。

```
Script started on Tue Mar 5 23:22:47 1997
```

```
cc2000(81)% date
```

```
1997年03月5日(水)23時22分51秒
```

```
cc2000(82)% uname
```

```
SunOS
```

```
cc2000(83)% cal 7 1999
```

```
7月 1999
```

```
日 月 火 水 木 金 土
```

```
1 2 3
```

```
4 5 6 7 8 9 10
```

```
11 12 13 14 15 16 17
```

```
18 19 20 21 22 23 24
```

```
25 26 27 28 29 30 31
```

```
cc2000(84)% exit
```

```
cc2000(86)%
```

```
script done on Tue Mar 5 23:23:19 1997
```

このように情報をファイルにする事によって、いつでも取り出せる形で保存する事ができるのです。一般的には、ファイルは明らかに消すという操作をしない限り消えることはありません。logout して、再び login しても、コンピュータの電源を切っても残っています。

#### ファイルの内容が長い場合

cat コマンドでファイルの内容を表示させた場合、その内容が長いと一画面に入り切らずにどんどんスクロール<sup>22</sup> して行ってしまいます。そのような場合には more コマンドで一画面ごとに表示を一旦停止させる事ができます。more ファイル名 <Return>です。

一旦停止している時に、以下のキーで様々なアクションを指示できます。

キー	アクション
<Space>	一画面分スクロール
<Return>	一行分スクロール
f	一画面分スキップして一画面分スクロール
b	二画面分戻って一画面分スクロール
h	どのキーを押せばどんなアクションをするかを教えてくれる

#### 4.4.5 ファイル名を変える

ファイル名は変更することができます。UNIX においてファイル名を変更するという作業は、実はファイルを別のところへ移動するということを意味します。つまり前のファイルは消えてなくなって、新しい名前生まれ変わるとい事です。コマンドは mv<sup>23</sup> です。引数が最低二つ必要で、最初の引数が元のファイル名、最後の引数が移動先のファイルです。例えば以下の例ではとりあえず test と付けたファイル名を log という名前に変えています。これはつまり test というファイルを log というファイルへ移動する、という事です。

<sup>22</sup> 新しい情報を表示するためにウィンドウ中の文字をずらすこと。表示しきれなくなった部分は画面から消える。

<sup>23</sup> move を縮めたつもりなのです

```
cc2000(86)% mv test log
```

test という名前のファイルがなくなって log という名前のファイルができた事を ls コマンドで確認しましょう。

```
cc2000(86)% ls
Apps Mail Wnn6 log
cc2000(87)%
```

#### 4.4.6 ファイルの複写

ファイルはその内容をそっくりそのままに複写することができます。コマンドは cp<sup>24</sup> です。引数が最低二つ必要で、最初の引数が元のファイル名、最後の引数が複写先のファイルです。例えば以下の例では log ファイルを log2 という名前で新しく作り、内容はそっくり log ファイルから引き写しています。先の mv との違いは、元のファイルがそのまま残る ( cp ) か、それとも消える ( mv ) かです。

```
cc2000(87)% cp log log2
cc2000(88)%
```

log2 という名前のファイルが増えた事を ls コマンドで確認しましょう。

```
cc2000(88)% ls
Apps Mail Wnn6 log log2
cc2000(89)%
```

#### 4.4.7 ファイルの消去

ファイルは消去することができます。逆に消去しない限りいつまでもそこに残っています。ファイルの置き場所は容量的には限りがあり、しかも複数の人で同じ置き場所を使っている場合が多いので、不要なファイルは削除するように心がけましょう。コマンドは rm<sup>25</sup> です。引数としてファイル名を与えます。例えば以下の例では log2 ファイルを削除しています。

```
cc2000(89)% rm log2
cc2000(90)%
```

log2 という名前のファイルが消えた事を ls コマンドで確認しましょう。

```
cc2000(88)% ls
Apps Mail Wnn6 log
cc2000(89)%
```

---

<sup>24</sup> copy を縮めたつもりなのです

<sup>25</sup> remove を縮めたつもりなのです

## 4.5 ファイルを編集する

今まで扱ってきたファイルは、先ほど script コマンドで作成したものです。ファイルは、このようにして作成するばかりでなく、利用者が自分の好きなようにその内容を変更したり、情報を追加したりできます。例えば先の script コマンドで作成したファイルも、レポートのつもりであれば自分の学部、学科、氏名を先頭に入れたいものです。

そこでファイルを自由に編集する方法を紹介します。Mule です。ここでは Mule<sup>26</sup> を使ってファイルを編集する作業を簡単に説明します。Mule はかなり多機能なソフトウェアで非常に多くの機能があります<sup>27</sup> が、ここではそれらのほとんどを説明しません。

### 4.5.1 Mule での作業の流れ

初めに Mule でファイルを編集するときの作業の流れを示しておきます。

- Mule の起動
- ファイル名の指定
- 編集
- ファイルの保存
- Mule の終了

Mule ではコントロールキーを押しながらの作業が非常に多くなります。念のため今一度説明しておきます。

コントロールキーを押しながら x キーを押すことをここでは C-x と表記します。 エスケープキーを押してから x キーを押すことをここでは M-x と表記します。
---

### 4.5.2 Mule の起動

Mule を利用する環境には二通りあります。

- A. X ウィンドウ環境が目の前にある場合。計算機センターが管理しているコンピュータ環境における具体例としては Ultra1 の前に座って利用している場合です。
- B. X ウィンドウ環境が目の前にない場合。計算機センターが管理しているコンピュータ環境における具体例としては、パソコンが置いてある情報処理教室から Enterprise3000(cc2000) を利用している場合などです。

いずれの場合も Mule でファイルを編集する場合のコマンド名は mule ですが、それぞれの場合に応じたすこし作法が変わります。以下に Mule を起動するところを別々に説明します。注意深く読んでください。図 4.2 に、最初にうまく Mule が起動できた場合の画面表示を示しておきます。

<sup>26</sup> 「みゅー」と読んでください。

<sup>27</sup> それを全て紹介した本の厚さは軽く 2 センチありますね。

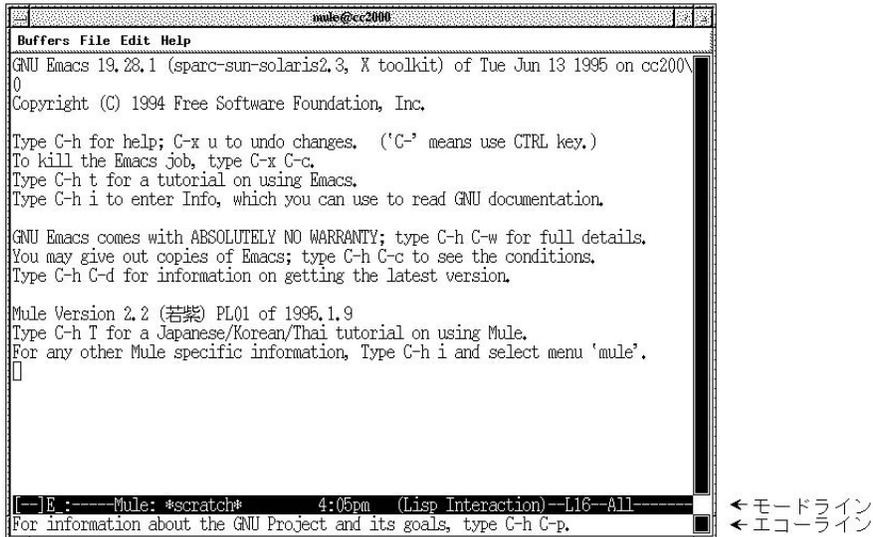


図 4.2 Mule の起動直後

Mule は起動すると常に最初の時点で画面内にメッセージを書いています。いつでも同じメッセージですので、特に気にしないでください。

Mule の画面をよく見ると、下から二行目に黒く色が反転した行がありますね。Mule は、この行を境にして、画面をそれより上の数十行の部分とそれより下の一行の部分に分けて使います。上から順番に以下のように呼ばれています。この呼び名は時々説明に現れますので、覚えてください。

- ウインドウ 編集するファイルの内容が表示される。(X ウインドウなどのウインドウと区別するために Mule のウインドウ、テキストウインドウなどと表現することもあります。)
- モードライン 編集しているファイルなどについての情報が表示される。常に白黒反転して表示される。
- エコーライン Mule からのメッセージなどが表示される

#### 4.5.3 A. の場合：X ウインドウ環境での Mule の起動とエラー対策

mule & <Return>で Mule を起動します。

```
csso1001(85)% mule &
[1] 9503
csso1001(86)%
```

新たなウインドウが一つ開いて、図 4.2 のような画面表示になると思います。もしもそうならない場合は、まず <Return>を一度押して、エラーメッセージがあるかどうかをチェックしてください。エラーメッセージを見て次のいずれの症状が発生しているかをよくチェックして、その対処を施し、もう一度 mule & コマンドを試みてください。

- ・エラーメッセージ : Please set the environment variable TERM; see tset(1).

このエラーメッセージが表示された場合は以下のコマンドを実行し、再度 `mule & <Return>` です<sup>28</sup>。

```
csso1001(86)% setenv DISPLAY unix:0.0
```

- ・エラーメッセージ : Xlib: connection to "unix:0.0" refused by server 他

```
Xlib: connection to "unix:0.0" refused by server
Xlib: Client is not authorized to connect to Server
X server not responding. Check your DISPLAY environment variable.
```

上記のようなエラーメッセージが出た場合はちょっと問題です。本当にあなたが利用している環境は X ウィンドウ環境ですか？あなたが該当するのは B. のケースではないですか？一度確認してください。B. のケースであれば B. のケースでのこのエラーの欄を参照してください。

もし本当に X ウィンドウ環境でこのようなことが起きるのなら、以下のコマンドを実行し、再度 `mule & <Return>` です。「`'`」バッククオートに気を付けてください。

```
csso1001(86)% setenv DISPLAY 'hostname':0.0
```

- ・エラーメッセージ : X server not responding. Check your DISPLAY environment variable.

上記のようなエラーメッセージが出た場合はちょっと問題です。本当にあなたが利用している環境は X ウィンドウ環境ですか？あなたが該当するのは B. のケースではないですか？一度確認してください。B. のケースであれば B. のケースでのこのエラーの欄を参照してください。

#### 4.5.4 B. の場合 : 非 X ウィンドウ環境での Mule の起動とエラー対策

`mule <Return>` で、Mule を起動します。

```
cc2000(85)% mule
```

これで図 4.2 のような画面表示になると思います。もしもこうならない場合は、以下のいずれの症状が発生しているかをよくチェックして、その対処を施し、もう一度 `mule` コマンドを試みてください。

- ・エラーメッセージ : Please set the environment variable TERM; see tset(1).

このエラーメッセージが表示された場合は以下のコマンドを実行し、再度 `mule <Return>` です。

```
csso1001(86)% set term=vt100
```

- ・エラーメッセージ : Xlib: connection to "unix:0.0" refused by server 他

```
Xlib: connection to "unix:0.0" refused by server
Xlib: Client is not authorized to connect to Server
X server not responding. Check your DISPLAY environment variable.
```

<sup>28</sup> UNIX に慣れた人はおやっと思われるかもしれませんが、一般的に X 環境下で DISPLAY 変数が設定されていれば TERM 変数の問題でトラブルは出ないのです。故にまず DISPLAY 変数が設定されていないことを疑うべきです。

上記のエラーメッセージが表示された場合は以下のコマンドを実行し、再度 `mule <Return>` です。上記の `"unix:0.0"` の部分が多少違っていても対処は同じです。

```
csso1001(86)% unsetenv DISPLAY
```

- ・エラーメッセージ : `X server not responding. Check your DISPLAY environment variable.`

上記のエラーメッセージが表示された場合は以下のコマンドを実行し、再度 `mule <Return>` です。上記の `"unix:0.0"` の部分が多少違っていても対処は同じです。

```
csso1001(86)% unsetenv DISPLAY
```

#### 4.5.5 ファイル名の指定

Mule が無事に起動できたら、まずは編集するファイルの名前を指定しましょう。指定した名前のファイルが存在しない場合、Mule はその名前で作成したファイルを作成します。指定したファイルが既存のものならば、Mule はそのファイルを読み込みます。

起動した直後の状態では、モードラインの左の部分に `*scratch*` と表示されているでしょう。ちょっと覚えておいてください。

ファイルを指定するためには `C-x C-f` です<sup>29</sup>。まずはじめに `C-x` を押してください<sup>30</sup>。数秒待つとエコーラインに `C-x-` と現れるでしょう<sup>31</sup>。ここで更に追い打ちを掛けるように `C-f` です。するとエコーラインには以下のように表示されますね。

```
Find File: ~/
```

この状態で編集したいファイル名をタイプし、そして `<Return>` です。今回は先ほど作成した `log` ファイルを編集する事にして、ファイル名のところに `log` とタイプしましょう。`<Return>` で `log` ファイルの内容が Mule のテキストウインドウ部分に読み込まれましたね。

さて、先ほどちょっと覚えておいてと言っていたモードラインの左の部分に `log` と表示されているのが判りますか？つまりモードラインのこの位置は、今現在編集しているファイルの名前を常に表示しているのです。最初に `*scratch*` だったのはまだ何のファイルも編集していないよということを示していたのです。

#### 4.5.6 編集

いよいよファイルの編集です。テキストウインドウには目標のファイルが読み込まれていますね。この状態で、カーソルは恐らくテキストウインドウの左上端だと思われます。この状態でタイピングすれば、カーソルのある位置にタイプした文字が入力されます。タイプした文字を消したい場合は `<Delete>` です。

#### カーソルの移動

カーソルは以下の方法でテキストウインドウの任意の位置に移動することができます。移動した先でタイプすればその文字がカーソルのある位置に入力されます。

<sup>29</sup> `C-x` の表記法に見覚えのない人は 4.1.2 を読み返してください。

<sup>30</sup> `C-x` の後で `<Return>` などしてはいけません。

<sup>31</sup> 実はこのエコーラインの表示を待つ必要はありません。慣れたら待たずに次のキーを押してください。

左矢印 ( ),C-b	カーソルを左に一文字移動
右矢印 ( ),C-f	カーソルを右に一文字移動
上矢印 ( ),C-p	カーソルを上に一文字移動
下矢印 ( ),C-n	カーソルを下に一文字移動
C-a	カーソルを行の先頭に移動
C-e	カーソルを行の末尾に移動
M-b	カーソルを左の単語に移動
M-f	カーソルを右の単語に移動
M-a	カーソルを文の先頭に移動
M-e	カーソルを文の末尾に移動

## 画面の移動

カーソルをどんどん下に動かして行けばいつかテキストウインドウの下端に到達します。そこで更にカーソルを下に移動させれば画面が一ページ弱スクロールします。カーソルキー以外にも画面を移動させる方法としては、以下のキーがあります。

C-v	画面を下に（つまり次の画面に）移動
M-v	画面を上（つまり前の画面に）移動
M-<	文頭（つまりファイルの先頭に）移動
M->	文末（つまりファイルの最後に）移動
C-1	（<Control>と英字の L です。）カーソルのある行を画面中央に持ってくるように画面を移動

## 行の移動

Mule でしばらく編集していると、ある行の前後関係を入れ換えたい、もしくは数行以上またいだ別の位置に移動したいと思うことがあるでしょう。このような場合は以下のステップを追うことで行の移動として実現できます。

1. 移動したい行を C-k で削除する。
2. 移動先にカーソルを移動させる。
3. C-y で 1. により削除した行を複写する。

つまり C-k を一度押すことによってカーソルの位置より右の一行分を削除することができます。C-k を続けて何度か押すことによって更に下の行もまとめて削除することができます。注意しなければならないのは、この削除の最中にカーソルを移動したり、何かほかの操作をしてはならないということです。あくまで連続した C-k の繰返しとして実行する必要があります。

移動する行の削除が済んだら、今度は移動先の位置にカーソルを移動させます。その後で C-y とすると、先ほど削除した行がその場所に割り込むように複写されます。

このようにしておこなう行の移動は、一行でも複数行でも構いません。C-y は何度でも行えますから、行の複製としても応用できます。C-k のすぐ後でカーソルを全く動かさずに C-y を実行すれば削除する前の状況に戻りますから、それからカーソルを移動してまた C-y をすれば、行を違う場所に複写することにも使えます。様々な応用が利く方法ですのでぜひ覚えてください。何度か失敗するかもしれませんが、機会を見てじっくり練習するのがお勧めです。

#### 4.5.7 ファイルへの保存

カーソルを自由に動かして自分の思うようにファイルを編集したら、ここで保存をしましょう。ファイルは Mule に読み込まれて Mule の中で編集されているだけで、元のファイルは全く編集されずにそのまま残っています。Mule の中の編集された結果を元のファイルに書き込むことによって、編集の結果が反映されるという仕掛けです。この種の操作を一般的に「保存」「セーブ (save)」と呼んでいます。

ファイルへの保存は C-x C-s です<sup>32</sup>。保存がうまく行けばエコーラインに

```
Wrote /NF/home/syokuin0/sheryl/log
```

などと表示されます。

#### 4.5.8 Mule の終了

さて、保存も済んだら Mule を終了したいところですが、再び 4.5.2 の場合分けに戻ります。

A. の場合だと、Mule は別の X のウィンドウとして起動されていますから、もともと Mule を起動したシェルが動いているウィンドウは別に残っており、そこでまた別のコマンドを実行できますから、特に Mule を終了する必要はないでしょう。シェルのウィンドウをクリックしてそちらをアクティブにするだけで良いですね。

B. の場合だと、Mule を終らないとシェルのプロンプトが現れず<sup>33</sup>、次のコマンドが実行できませんから Mule を終らなければなりません。

いずれにしても Mule を終了するには C-x C-c です。もしもまだ保存していないファイルを編集集中に Mule を終了しようとした場合は、以下のようなメッセージがエコーラインに現れます。

```
Save file /NF/home/syokuin0/sheryl/log? (y ,n, !, ., q, C-r or C-h)
```

編集中のファイルを保存して Mule を終了する場合は y をタイプします。それでシェルのプロンプトが現れるでしょう。

保存しない場合は n をタイプします。この場合、Mule は念のためにもう一度以下のような確認の問い合わせをします。

```
Modified buffers exist; exit anyway? (yes or no)
```

今度は yes<Return>とタイプします。これで編集集中の内容はファイルに保存されず、シェルのプロンプトが現れるでしょう。

#### 4.5.9 Mule もっともっと

ここまでで非常に簡単に Mule の使い方を紹介してきました。でもこれだけの機能で日常的にファイルを編集するのはやはり不便があると思います。

実際、Mule にはもっともっとさまざまな機能があります。ここではさらに深く Mule の使い方を知りたい人のためにいくつかの方法を紹介しましょう。

<sup>32</sup> くだいですが <Return>キーなど押さずにコントロールキーを押しながら x s と押します。

<sup>33</sup> 実はそんな事はなくて Mule の中からシェルを呼び出すこともできるのですが、ここでは説明しません。

## その前にちょっとしたテクニック

Mule は非常に多くの<Control>キーや <ESC> キーを利用した機能があります。これらの機能を実行しようとして、キー操作を間違えてしまったり、操作の途中でおかしくなってしまった場合、下手をするとどんどんと深い失敗の谷に落ちて行くときがあります<sup>34</sup>。そういう事にならないためには、やはり「あぶない」と思ったときにはすぐに作業を中断するのが得策です。Mule では作業の中断は一般的に C-g です。何が起きてこれさえ覚えておけばもうそれ以上失敗の傷を広げることはありません。覚えておくといいでしょう。

## もっと詳しいドキュメント

7 「Mule」に、より詳しい Mule の使い方についての説明がありますのでそちらを参照してください。また、付録の参考文献にも幾らか挙げておきますので、そちらも参照してください。

## 日本語をタイプしたい

今まではアルファベットをタイプする方法しか説明しませんでしたから、普通の人が日常的な言葉として使っている日本語を入力する事はできませんね。でも安心してください。Mule を利用して日本語も入力できます。これをコンピュータの世界では、かな漢字変換と呼びます。7.5 章に、Mule 上でのかな漢字変換機能についての説明がありますのでそちらを参照してください。

## チュートリアル

Mule には自己学習の為の機能がついています。図 4.2 に出ている表示をよく読めば判るように、Mule を起動した後、C-h T<sup>35</sup> とすれば Language: と表示されます。そこで Japanese とタイプして<Return>してください。ここで先頭の J が英大文字である事に注意！シフトキーを押しながら J キーを押します。後はシフトキーを離して apapanese です。以下のような画面表示になると思います。



図 4.3 Mule 入門 (部分)

この入門編を読みながらそこに書かれている通りに操作をしていけば、Mule の殆どの機能について知ることができます。(もしも入門編が英語で表示されたら先ほどの Japanese の綴りを間違えたのでしょう。一旦終わってからやり直すのがお勧めです。)この入門編を終るには(勿論終わり方も入門編に書いてありますが)4.5.8 を読んでください。入門編を始めた時のモードラインを見れば判るのですが、TUTORIAL.jp という名前のファイルを編集している事になっています。その為、終了作業をすると以下のように保存するかどうかを問い合わせてくる場合があります。

<sup>34</sup> これが実は結構怖いんです。

<sup>35</sup> T は大文字の T です。つまりシフトキーを押しながらアルファベットの T キーを押します。

Save file /NF/home/syokuin0/sheryl/TUTORIAL.jp? (y or n)

ここで 4.5.8 の記述にしたがって保存しておくこのファイルが残ってしまいます。特に必要でない限りはここでは保存しない方がいいでしょう。チュートリアルの中に保存したとしても、もう要らないと思ったらこまめに `rm TUTORIAL.jp <Return>` で消去するのがおすすめです。

## 4.6 印刷

UNIX 環境では様々なものをプリンタから印刷する事ができます。cc 環境でも、幾つかプリンタを用意しており、これは cc 環境を利用できるコンピュータからならどこからでも誰でも利用できるものです。

### 4.6.1 どんなプリンタがあるか

cc 環境から利用できるプリンタは以下のとおりです。プリンタは複数あるので、それぞれ名前が付けてあります。

プリンタ名	設置場所
apr2101 ~ apr2107	10 号館 2 階 10201 情報処理教室
apr2201 ~ apr2208	10 号館 2 階 10202 情報処理教室
apr2301 ~ apr2304	10 号館 2 階 10203 情報処理教室
apr2401 ~ apr2404	10 号館 2 階 10204 情報処理教室
apr2501 ~ apr2504	10 号館 2 階 10205 情報処理教室
apr2601 ~ apr2604	10 号館 2 階 10206 情報処理教室
apr3101 ~ apr3105	10 号館 3 階 10301 情報処理教室
apr3201 ~ apr3207	10 号館 3 階 10302 情報処理教室
apr3301 ~ apr3302	10 号館 3 階 10303 情報処理教室
apr4501	10 号館 4 階 10405 情報処理教室
apr4601	10 号館 4 階 10406 情報処理教室
cspr01,cspr02	2 号館 4 階 21 情報処理教室
clpr03 ~ clpr07	3 号館 1 階 32 情報処理教室
cepr01	5 号館 1 階 51 情報処理教室
c1kpr01	第 1 研究室棟 2 階共同研究室
c2kpr01	第 2 研究室棟 1 階共同研究室
c3kpr01	第 3 研究室棟 1 階共同研究室
cgpr01 ~ cgpr04	1 号館 2 階 11 情報処理教室
c9pr01	9 号館

初めてプリンタを利用するときは、まずそのプリンタのある場所まで行って、実際どのプリンタに出力されるのか確認しておく事をお勧めします。上の表は恒久的なものではなく、様々な要因でプリンタは配置換えをしたり名前を変更したりされる可能性があります。この種の状況の変化については 4.7 に説明している ccinfo コマンドを利用して知ることができます。後で参照してください。

プリンタの名前はそれぞれのプリンタ自体に貼り付けてありますから、まずプリンタのところに行って、プリンタの名前を確認しておくことがトラブルを起こさないためには重要です。

### 4.6.2 ファイルの印刷

lpr -P プリンタ名 ファイル名 <Return>とすれば指定のプリンタに指定のファイルの内容を印刷する事ができます。

```
cc2000(81)% lpr -Pcspr01 log
```

-P オプションに続くプリンタ名の指定は、上記の例のようにくっつけて書いてください。

例えば `-P cspr01` などとしてはいけません。第二の引数となるファイル名とプリンタ名の間には一つ以上の空白を入れてください。

## 注意

上記の方法でファイルを印刷する場合、どんなファイルでも印刷できるとは限りません。cat コマンドなどで内容が確認できるような文字ばかりのファイルに限ります。それ以外の、cat したら画面に変な文字がいっぱい表示されるようなファイルは印刷しないでください。プリンタが止まったり無駄に数百ページ印刷されたりします。

### 4.6.3 印刷状況をチェックする

`lpq -P プリンタ名 <Return>`で指定のプリンタの現在の状況の確認ができます。プリンタの状態を表すメッセージが何行か返ってきます。

#### プリンタがすいている状態

以下はプリンタに何も出力待ちのものが無い場合の例です。

```
cc2000(82)% lpq -Pcspr01
cc2000: Tue Mar  8 13:48:02 1994:
no entries
cc2000(83)%
```

上記のようではなく、単に `no entries` メッセージだけが返ってくる場合もあります。

#### プリンタがなにかを印刷している状態

以下のようなメッセージが返ってきたら、それはプリンタが何か印刷している最中か、もしくは印刷のための準備中だということです。

```
cc2000(88)% lpq -Pcspr01
cc2000: Fri Mar 11 15:29:59 1994:
cspr01 is ready and printing
Rank  Owner      Job  Files                Total Size
active tanaka   43   sample.ps           152 bytes
1st   yamada     46   test.text            8 bytes
cc2000(89)%
```

上記の例では tanaka さんの sample.ps というファイルが現在印刷中 (active) で、その次 (1st) の yamada さんの test.text というファイルが印刷待ちだということです。

ユーザ ID の右に出ている 43 や 46 の数字は印刷要求それぞれに割り当てられた番号で、ジョブ番号と呼ばれています。

#### 4.6.4 印刷の取消し

一旦 `lpr` コマンドでプリンタに流し込んだ出力要求を取り消して、印刷しないようにできます。 `lprm -P` プリンタ名 ジョブ番号 `<Return>` です。一つ上の例の、yamada さんは慌て者で、間違っ別別のファイルを印刷するようにコマンドを実行してしまいました<sup>36</sup>。この yamada さんの印刷要求に付けられたジョブ番号は 46 です。これを取り消す例を示します。

(自分以外の印刷要求は決して取り消すことはできません。)

```
cc2000(99)% lprm -Pcspr01 46
benkei.kyoto-su.ac.jp: dfA046cc2000 dequeued
benkei.kyoto-su.ac.jp: cfA046cc2000.kyoto-su.ac.jp dequeued
cc2000(100)%
```

実際この例とはかなり違ったメッセージが表示されるかもしれませんが、重要なのは `dequeued` で、このメッセージが表示されればまず間違いなく印刷要求は取り消されています。念のためにもういちど `lprm -Pcspr01 <Return>` などして印刷要求が消えていることを確認するのがいいでしょう。

#### 4.6.5 利用上の注意

ちょっとひとこと

最近プリンタの紙の無駄遣いや古紙の散乱が目立ちます。プリンタに印刷しようと思ったら、紙は使い切っているわ、周りは古紙だらけで汚いわ、では困ります。印刷するなというつもりはありませんが、慣れるに従ってお互い無駄な印刷をしない様にし、また積極的に散乱しているプリント結果の整理整頓をお願いします。共有して利用するものへの愛情を忘れずにしてください。利用者全体のモラルが問われます。

プリンタのトラブル

プリンタを使っていると、時々紙詰まり(ジャムと呼ばれる)や印刷が薄いなどのトラブルが発生します。この種のトラブルが発生したら計算機センターまで連絡ください。また、MiCS 補助員(計算機運用補助員)と呼ばれる学生が各情報処理教室を回っていますから、彼らに頼むのもいいでしょう。紙やトナー(インクのようなものです)の補給も彼らが行います。

---

<sup>36</sup> こんな人が紙を無駄遣いするんですね。

## 4.7 状況の変化

cc 環境は全く固定的な環境では無く、常に変化しています。例えば先に挙げたプリンタの配置や名前は、新しいプリンタの導入や利用者の要求に応じて変化して行きます。このドキュメントは印刷物ですから、印刷した時点で固定されてしまいます。最近に起こった変化は反映されていない場合もあるでしょう。

それを解決するために cc 環境では ccinfo というコマンドを用意しています。これは cc 環境で配布されているドキュメントや様々な情報の最新のものを利用者が簡単に取り出せる事を目標に設定されました。

ccinfo<Return>で起動できます。以下のようなガイドメニューが表示されるでしょう。この ccinfo コマンドを起動した直後のメニューをトップメニューと呼んでいます。例えばプリンタの情報は「1. cc 環境の設備について」以下にあります。

```
cc2000(125)% ccinfo
```

-----  
ここでは一般的な情報の検索が可能です。

まずは分野を選んでください。

- 1 cc 環境の設備について
- 2 各種ドキュメント
- 3 ニュースグループについて
- 4 ネットワーク関係の情報
- 5 各種連絡先
- 6 そのほかの話題

番号を入力してください( 0 で終了します) :

ccinfo コマンドは基本的にはメニューに表示される項目について、その番号をタイプすることによって選択する様になっています。番号として 0 (ゼロ) を入力すると一つ前のメニューに戻ります。トップメニューで 0 を入力すると ccinfo コマンドそのものを終了します。

ccinfo コマンドのメニュー内容は常に更新され、その機能も常に変化していくでしょう。しかしどのように変化したとしても、基本的にはメニューの番号を選ぶか、問い合わせに対して y または n など返事をする事によって操作が進行するように作られています。

様々な情報が掲載されていますので、一度覗いてみることをお勧めします。

## 4.8 ファイルの階層構造

これまでで一般的なファイルの操作について説明してきました。ところで UNIX ではファイルは階層化されています。ここでは階層化されたファイルの概念と扱い方を説明します。

### 4.8.1 ディレクトリ

コンピュータをしばらく使っていると結構ファイルが増えて来て、そのうちどのファイルが何のためのものだったのか分からなくなるものです。ls コマンドで一覧を見たら一画面では収まり切らなくなったりして大変な状況の人も出てくるでしょう。そうならないために例えばファイルの名前を長くして、その名前を見ればファイルの内容の想像が付くようにするなど利用者は色々な対策を取ったりします。しかしそういう手法で全てが解決するわけでもありませんね。

4の4.4で、UNIX が扱うファイルはあなたの机の上に並ぶファイルのようなものだと言いました。もしもあなたの机の上にファイルが非常にたくさん並びはじめ、ファイルの背表紙にちょっと長めの名前を書いておくくらいでは目的のファイルがどこにあるのかすぐに探せなくなってしまったとしましょう。あなたはどうしますか？

多くの方は「整理が必要だ」と感じるでしょう。ファイルを分類し、分類ごとに大きな区分を作ってファイルを束にして置いておけば良いというわけです。分類とは物事を階層化して整理するということです。まず大分類があって、それから中分類、更に必要なら小分類、という感じですね。以下に日常的な机の上の分類の例を示します。四角で分類、丸でファイルを表しています。

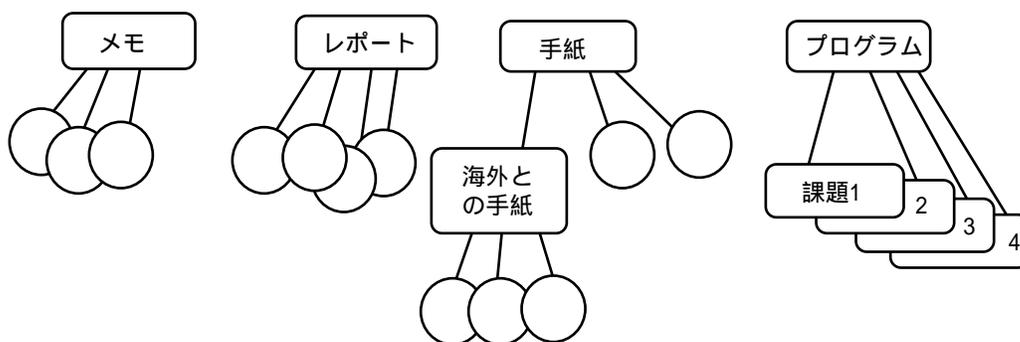


図 4.4 階層化された分類の例 (1)

UNIX でもファイルを階層化して整理することが可能です。以下に先の例に合うような階層化を行ったファイルの配置の例を示します。

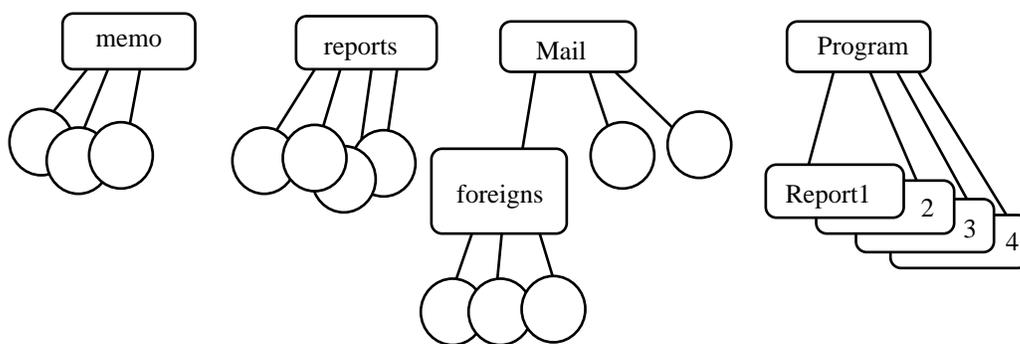


図 4.5 階層化された分類の例 (2)

しかし実は UNIX コンピュータは yasuda さん一人のものではなくて、数多くの人が同時に使っているということを忘れてはいけません。コンピュータにとっては、「yasuda のファイル」という分類がまず最初に既に存在しているのです。

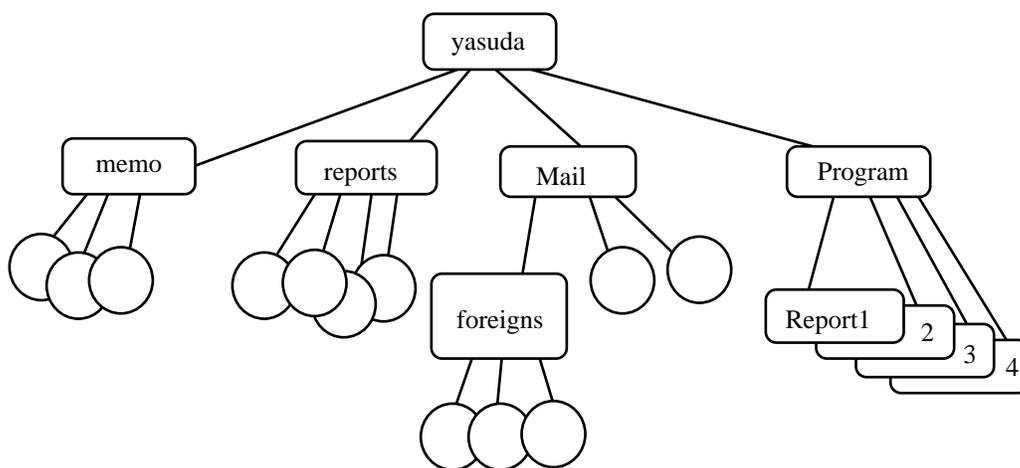


図 4.6 階層化された分類の例 (3)

そして「yasuda のファイル」は以下のようにもっと大きな分類の下に配置されているのです。

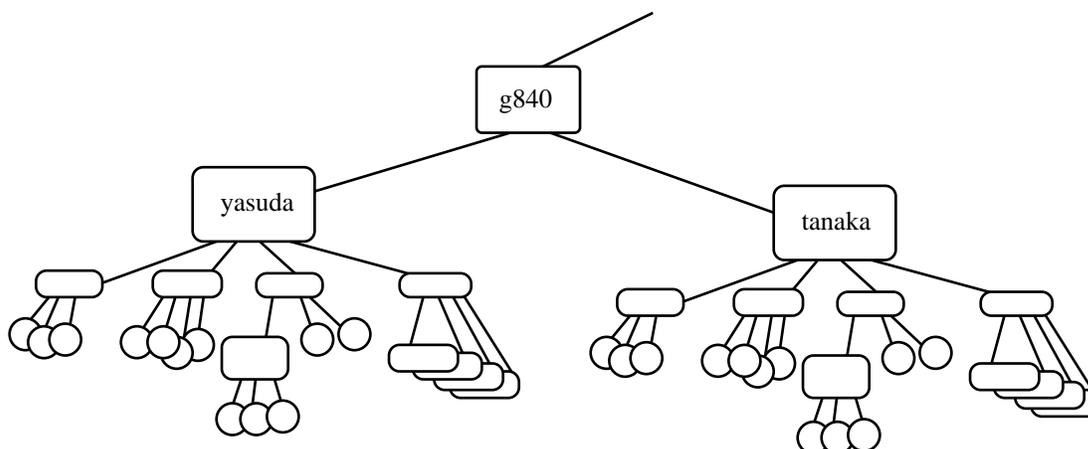


図 4.7 階層化された分類の例 (4)

上の例での「yasuda のファイル」より一つ大きい分類の名前は「g840」です。これは「ある年度に入学してきた学生のファイル」という意味の分類です。これよりもう少し上の存在として「全ての利用者のファイル」というものが有り、更にたどって行くと最終的に「全てのファイル」という最大の分類に到達します。

先ほどの例では四角で表してきた、ファイルの分類なるものの存在を UNIX では「ディレクトリ<sup>37</sup>」と呼んでいます。そして、先ほどの「全てのファイル」という最大の分類、つまり分類の根本（もしくは頂上）なるものの存在を「ルートディレクトリ<sup>38</sup>」もしくは「トップディレクトリ」などと呼んでいます。

「yasuda のファイル」は実は yasuda という名前が付いたディレクトリなのです。この、利用者ごとに割り当てられた利用者自身のためのディレクトリを「ホームディレクトリ」と呼んでいます。ホームディレクトリは常に利用者のユーザ ID が付けられて、利用者登録の際にシステム管理者<sup>39</sup>の手によって作成されます。利用者は自分のホームディレクトリ以下に自分のファイルもしくはディレクトリを作り、保存することができます。

いくつかのディレクトリを挙げましたが、いずれにしてもディレクトリには違いありません。

UNIX のディレクトリとファイルの構造を見ると、それが木の根の構造のように見えると思います。また上下をひっくり返せばルートディレクトリを根にしてディレクトリの分類過程が枝のように、そしてファイルが葉のように見えるかもしれません。このようなイメージで表せる構造のことをコンピュータの世界では「木構造」「ツリー (tree) 構造」と呼んでいます。ディレクトリの木構造をディレクトリ・ツリーなどと呼んだりします。

ところで UNIX ではディレクトリはファイルの一種として扱われます。(扱われる、のです。普通のファイルとディレクトリはその意味に違いがあります。) そのため「ディレクトリファイル」などと表現されるときもあります。ディレクトリもファイルと同じ様に名前を付けてそれを他のものと区別して扱います。名前のルール（名前に利用できる文字、文字数の制限など）は普通のファイルと同じです。

<sup>37</sup> directory : 住所録？何故この呼び名を採用したのか、私は知りません。

<sup>38</sup> root directory : 根、ですね。

<sup>39</sup> cc 環境におけるシステム管理者は計算機センターです

## 4.8.2 ツリー構造におけるファイル名の表記

ファイルにはファイル名が有ることは説明しました。しかし UNIX のディレクトリ・ツリーの中で名前によってそのファイルを指定するにはツリー構造を含めて表現できる方法が必要ですね。

### 絶対パスによるファイル名の表記

例えばユーザ ID yasuda さんのホームディレクトリの名前は yasuda です。このディレクトリファイルを、UNIX コンピュータ全体のツリー構造の中では「/NF/home/g840/yasuda」などと表現します<sup>40</sup>。これが正にディレクトリ・ツリーの中で絶対的な位置と名前を表す表記法です。

先頭の「/」はルートディレクトリを表しています。それ以降の「/」はディレクトリ構造の区切り、つまり図 4.7 でのディレクトリとディレクトリもしくはファイルを結ぶ「線」に相当します。「/」には含まれた名前は全て途中に存在するディレクトリの名前であり、最後の名前はディレクトリ、もしくはファイルの名前です。

こうして表現することによってディレクトリ・ツリーの中で、名前によって完全にファイルが指示できるようになります。これを「絶対パスによるファイル名の表記」などと呼んでいます。絶対パスによる表記の場合、その表記はルートディレクトリを起点にして、たどって行くディレクトリの道のり (path : パス) を表現しているとみなせます。

### 相対パスによるファイル名の表記

でも絶対パスによってしかファイルの名前が表現できないとしたらこれは非常に不便なことです。例えば yasuda さんが自分のホームディレクトリの直下にある log というファイルを old-log という名前に変えたいと思ったときに、こんな風にコマンドを書かなくてはなりません。

```
cc2000(82)% mv /NF/home/g840/yasuda/log /NF/home/g840/yasuda/old-log
```

いやこれはたまりません。これでは単に長い名前をファイルに付けているようなもので、何のためにディレクトリという概念を導入したのか判りません。

そこでファイルを表現するのに、ディレクトリ・ツリーの前半部分をタイプしなくても済むように覚えておいて、ツリーの残りの部分だけ表現すればいいような表記方法があります。そのために UNIX のシェルは、常にディレクトリ・ツリーのどれか一つのディレクトリに注目しています。そのディレクトリまでのツリーの記述は省略可能となるわけです。例えば先の例の mv において、今注目しているディレクトリがホームディレクトリだったとすると、以下のようにコマンドを短く書くことができます。

```
cc2000(82)% mv log old-log
```

この「今注目しているディレクトリ」もしくは「今省略可能であるディレクトリ」を「カレントディレクトリ<sup>41</sup>」もしくは「ワーキングディレクトリ」と呼んでいます。カレントディレクトリはコマンドによって変更することができます。(後述)

pwd コマンドでカレントディレクトリを確認する事ができます。

```
cc2000(81)% pwd
/NF/home/g840/yasuda
cc2000(82)%
```

<sup>40</sup> あなたのホームディレクトリはあなたのユーザ ID が使われているはずですが、実際のあなたのホームディレクトリの名前が知りたければ login 直後に pwd コマンドで確認できます。

<sup>41</sup> current directory : 現在のディレクトリ

先の例の、省略された mv コマンドのファイルに関する表記では、この /NF/home/g840/yasuda が省略されていた<sup>42</sup> というわけです。

ところで、login 直後のカレントディレクトリは常にホームディレクトリです。つまり今まで 4 の 4.4 などでも試してきたファイルは、この省略された表記法によって表現された、あなたのホームディレクトリ直下にあったファイルだったのです。ls コマンドなどでその一覧が表示されていたのも、あなたのホームディレクトリ以下の内容だったというわけです。

こうして表現することによってディレクトリ・ツリーの中で、簡単な表記によってファイルが指示できるようになります。これを「相対パスによるファイル名の表記」などと呼んでいます。相対パスによる表記の場合、その表記はカレントディレクトリを起点にして、たどって行くディレクトリの道のり (path : パス) を表現しているとみなせます。

#### パスによる表記でもう少し

言い遅れましたが「絶対的な表記」と「相対的な表記」の区別は、その表記の先頭が「/」であるか否かで判断されます。

また、パス中には以下の記号が利用できます。

記号	意味
.	カレントディレクトリを意味します 例えば ./sample と書けば、カレントディレクトリにある sample というファイルを意味します。つまり単に sample と書いたのと同義です。(しかしそう書いたのでは駄目な場合もあるので。)
..	一つ上のディレクトリを意味します。 例えば ../sample と書けば、カレントディレクトリの一つ上のディレクトリにある sample というファイルを意味します。 例えば ../../sample と書けば、カレントディレクトリの二つ上のディレクトリにある sample というファイルを意味します。

また、シェルからコマンドの引数としてファイルを記述するとき、パスの先頭であれば以下のような書き方もできます。(但し sh では駄目です。tcsh か csh で有効です。)

記号	意味
~	自分のホームディレクトリを意味します
~username	ユーザ ID username のホームディレクトリを意味します

### 4.8.3 ディレクトリの扱い

#### ディレクトリの作成

ディレクトリを作成するには mkdir<sup>43</sup> コマンドを利用します。書式は以下のとおりです。

mkdir ディレクトリ名...

<sup>42</sup> 厳密には pwd の結果の最後にもう一つ / を付けないといけませんね。

<sup>43</sup> make directory の略なのです。

## ディレクトリの消去

ディレクトリを消去するには `rmdir`<sup>44</sup> コマンドを利用します。書式は以下のとおりです。

```
rmdir ディレクトリ名...
```

ディレクトリの消去は、そのディレクトリより下にディレクトリまたはファイルが含まれていてはできません。消去したいディレクトリ以下のファイルまたはディレクトリを `rm` または `rmdir` コマンドで予め消しておいてください。

## カレントディレクトリの表示

カレントディレクトリを表示するには `pwd`<sup>45</sup> コマンドを利用します。書式は以下のとおりです。

```
pwd
```

## カレントディレクトリの変更（移動）

カレントディレクトリを変更するには `cd`<sup>46</sup> コマンドを利用します。書式は以下のとおりです。

```
cd [ディレクトリ名]
```

login した直後はカレントディレクトリは常にホームディレクトリです。それから `cd` コマンドでどこかのディレクトリに移動しても、単に `cd<Return>`（つまり引数であるディレクトリ名を省略）とすると、常にホームディレクトリに移動します。

### 4.8.4 ディレクトリを意識したコマンドの書き方

今まで紹介してきたファイルを扱うコマンドは、そのほとんどがディレクトリに対しても適用できます。これ以降に以下の図の状況を例に取って説明します。

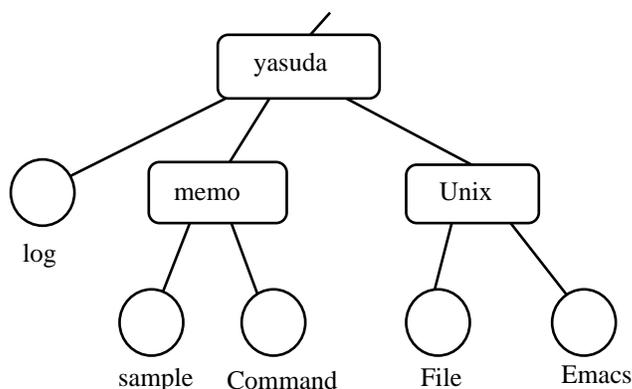


図 4.8 ディレクトリの例

<sup>44</sup> remove directory の略なのです。

<sup>45</sup> print working directory の略なのです。

<sup>46</sup> change directory の略なのです。

## まずは練習材料を作る

まず初めに練習のために図 4.8 の状況を作るのがおすすめです。以下にその手順を示します。今のカレントディレクトリはホームディレクトリではないかもしれませんが念のために最初に `cd<Return>` してホームディレクトリに移動しておきましょう。

```
cc2000(84)% cd
cc2000(84)% mkdir memo Unix
cc2000(84)% ls
Apps Mail Unix Wnn6 log memo
cc2000(84)%
```

## ファイルのコピー

`cp` コマンドには以下の 3 通りの書き方があります。

1. `cp [-i] file1 file2`
2. `cp [-i] file... dir`
3. `cp -r dir1 dir2`

`-i` オプション<sup>47</sup> を与えると、コピーする際に同名のファイルが既に存在し、コピーすることによって上書きされて元の内容がなくなってしまうような場合に実行してよいかどうか問い合わせを行う。これに `y` と答えると実行し、それ以外の入力であれば実行しない。

1. の書き方では、`file1` は `file2` に単にコピーされます。
2. の書き方では、(もし複数書けば複数の) `file` は、`dir` のすぐ下に元のファイル名でコピーされます。
3. の書き方では、`dir1` 以下のファイルを全て含めてディレクトリごと `dir2` のすぐ下に元のファイル名、ディレクトリ名でコピーされます。`dir2` が存在しないとそれを作成して、`dir1` 以下のファイルを全て `dir2` のすぐ下に元のファイル名、ディレクトリ名でコピーされます。

例えば `log` ファイルを `memo` ディレクトリ以下に同じく `log` という名前でコピーする場合、以下のようなささまざまな表現ができます。

カレントディレクトリ	書き方	コマンド記述
yasuda	1.	<code>cp log memo/log</code>
yasuda	2.	<code>cp log memo</code>
memo	1.	<code>cp ../log log</code>
memo	1.	<code>cp ../log ./log</code>
memo	2.	<code>cp ../log .</code>

---

<sup>47</sup> inquiry 問い合わせ、のつもりでしょうか

例えば sample ファイルを Unix ディレクトリ以下に同じく sample という名前でコピーする場合、以下のようなさまざまな表現ができます。

カレントディレクトリ	書き方	コマンド記述
yasuda	1.	cp memo/sample Unix/log
yasuda	2.	cp memo/sample Unix
memo	1.	cp sample ../Unix/sample
memo	2.	cp sample ../Unix

例えば memo ディレクトリをツリーごと Unix ディレクトリ以下にコピーする場合、以下のようになります。(今度は書き方は 3. しかありません。)

カレントディレクトリ	コマンド記述
yasuda	cp -r memo Unix
memo	cp -r ../memo ../Unix (どういうわけか cp -r . ../Unix は駄目なのです)
Unix	cp -r ../memo .

この結果、ディレクトリ・ツリーは以下のようになります。

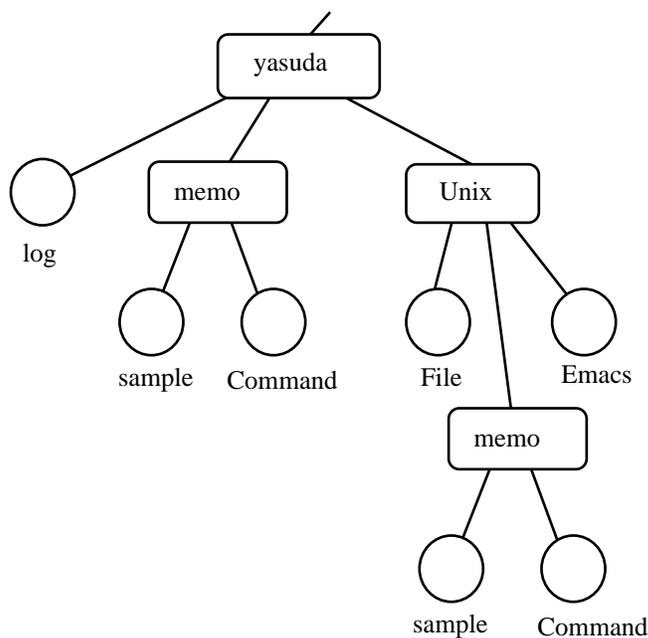


図 4.9 コピー後のディレクトリ・ツリー

## ファイルの移動

mv コマンドには以下の 3 通りの書き方があります。

1. mv [-if] file1 file2
2. mv [-if] file... dir
3. mv dir1 dir2

-i オプション<sup>48</sup> を与えると、移動する際に同名のファイルが既に存在し、移動することによって上書きされて元の内容がなくなってしまうような場合に実行してよいかどうか問い合わせを行う。これに y と答えると実行し、それ以外の入力であれば実行しない。

-f オプション<sup>49</sup> を与えると上書きされて元の内容がなくなってしまうような場合でも問い合わせせず実行する。

1. の書き方では、file1 は file2 に単に移動されます。
2. の書き方では、(もし複数書けば複数の) file は、dir のすぐ下に元のファイル名で移動されます。
3. の書き方では、dir1 以下のファイルを全て含めてディレクトリごと dir2 のすぐ下に元のファイル名、ディレクトリ名で移動されます。dir2 が存在しないとそれを作成して、dir1 以下のファイルを全て dir2 のすぐ下に元のファイル名、ディレクトリ名で移動されます。

mv コマンドはすぐ判るように cp コマンドとほとんど同じ記述方法ができます。働きもほとんど同じで、単に元のファイルが残る (cp) か残らない (mv) かだけです。cp コマンドと同じですので、ここでは例を挙げません。

最後の 3. の書き方で memo ディレクトリを Unix ディレクトリにディレクトリ・ツリーごと移動した場合は、図 4.9 に挙げた例の左側の元の memo ディレクトリと、それ以下の sample と Comand ファイルがなくなった状態になります。

## ファイルの名前を変える

ファイルの名前の変更は mv コマンドで行ないます。つまり sample というファイルの名前を otamesi に変えるという作業は sample ファイルを otamesi に移動するという事と考えるのです。具体的には単純に mv sample otamesi<Return>です。

## ファイルの消去

rm コマンドは以下の書き方をします。

```
rm [-ifr] file...
```

-i オプション<sup>50</sup> を与えると、消去する際に場合に実行してよいかどうか問い合わせを行う。これに y と答えると実行し、それ以外の入力であれば実行しない。

---

<sup>48</sup> inquiry 問い合わせ、のつもりでしょうか

<sup>49</sup> force 強制、のつもりでしょうか

<sup>50</sup> inquiry 問い合わせ、のつもりでしょうか

-f オプション<sup>51</sup> を与えると問い合わせせず実行する。  
-r オプション<sup>52</sup> を与えて file の部分がディレクトリだった場合は file 以下のファイルを全て含めてディレクトリ・ツリーごと消去する。

特に rm コマンドに-r を指定してディレクトリ・ツリーごと消去するという例は非常に有用です。なにしろディレクトリの消去である rmdir コマンドは、そのディレクトリ以下の内容が空になっていなければいけないのです。そのような場合は rm -r で一発消去ができます。

---

<sup>51</sup> force 強制、のつもりでしょうか

<sup>52</sup> recursive 回帰的、のつもりでしょうか

## 第5章 ネットワークの世界へようこそ

ここしばらくの間でインターネットの知名度は非常に上がりました。技術者では無く、利用者の眼から見たインターネットとは、「インターネットを使ってできる事」の集合体です。例えば電子メールや WWW<sup>1</sup> が最も代表的な二つ<sup>2</sup> でしょうか。この様な「できる事」をネットワークサービスと呼んでいます。

あなたも cc 環境の利用者として、無数のネットワーク・サービスを利用することができます。ここではそれらのうち、より一般的なものを取り上げてその概要と使い方を説明します。

### 5.1 ネットワークサービス紹介

ここでは代表的なネットワークサービスとして、「電子メール」「ネットニュース」「WWW」がどのようなものかを説明します。既に知識と経験のある人は 118 ページ、5.3 まで読み飛ばしてください。但し 115 ページ、5.2.2 については読んでおく方が良いかもしれません。

#### 5.1.1 ネットワークサービスと向き合う

「電子メール」や「WWW」とは何なのかを説明するより前に、ネットワークサービスと人との向き合い方に関する話をしましょう。

様々なネットワークサービスの中で、最も重要なサービスは恐らく「電子メール」です。最近のインターネットの流行と、その大きな原動力となっている「WWW<sup>3</sup>」を無視する訳ではありません。しかし、それでも「電子メール」という、この最も古くからあるネットワークサービスが、今でも、そして今後も最も重要なサービスであることに間違いは無いと考えています。

それは、ネットワークの中で様々な情報を交換し、WWW の中身を含めて様々なものを構築しているのが人間だからです。

人と人とのネットワーク越しの情報交換が、ネットワーク・サービスの本質なのです。電子メールや WWW は、それらの人達の活動を支える単なる道具に過ぎないと言っても良いでしょう。その、ネットワーク越しに行なわれる人と人との情報交換の最も基本的な形が、電子メールなのです。故に電子メールが今後も最も重要なネットワーク・サービスであり続けると考えているのです。

もちろん電子メールを使えなくても WWW を覗くことはできますが、それではあなたは単なるネットワーク社会の観客に過ぎません。ガラス越しに、別の部屋からネットワーク社会をただ眺めているのです。

ネットワークは人と人がメッセージを交換する「場」なのです。ただ受けとるだけで誰とも情報を交換しない、ただの観客となるのは詰まらないことです。勿論その逆に言っぱなし、書きっぱなしになるのも困ります。一方通行ではない情報の交換ができてはじめて一人前のネットワークです。ネットワーク社会の住人達が同じ対等な人として向き合うための道具として、電子メールは使われています。逆に電子メールが使えるようになるということは、自分が一人前のネットワーク住人となるための第一歩でもあ

<sup>1</sup> 「だぶりゅーだぶりゅーだぶりゅー」と読むのが妥当なようです。World Wide Web の略。単に「ウェブ」と呼ばれる場合もあります。「だぶだぶだぶ」などと読む人も居ます。

<sup>2</sup> もちろんインターネット関連の変化は急激で、明日 WWW などより有名になる何かが現れるかもしれません。

<sup>3</sup> 雑誌やテレビなどで「インターネットのホームページ」などと呼ばれているものです。

るのです。

cc 環境では、その利用者は誰もがネットワーク社会の住人となります。ネットワーク越しに多くの人とメッセージを交換し、様々なことを実現できるのです。あなたも cc 環境ネットワークの、そしてインターネットの住人になるべく、まず電子メールの読み書きができるようになるうではありませんか。ネットニュースでコミュニティに参加しましょう。WWW で情報発信をしましょう。それはただ人が作った WWW ページを眺めて回るより 100 倍楽しいことなのです。

次節から「電子メール」「ネットニュース」「WWW」の概要と使い方を、順を追って説明していきます。

### 5.1.2 電子メールって何？

電子メールは通常の郵便システムと似た機能を持ちます。大きな相違は紙や物を運ぶのではなく、文字をデータ化して相手に届けるという点です。つまり通常の葉書や封書による郵便は、実際には紙としての葉書や物としての封書を届けるのが第一の目的では無く、そこに書かれた内容、つまり文字を相手に届けることが本質です。しかし実際には紙や物を届けることによって実現しているというのが郵政省の郵便システムと言えます。

それに対して電子メールでは、一般的に文面をコンピュータに入力する事によってデータ化し、コンピュータの機能を使って相手に送り付けます。受け取った側でも、データ化された文書を、受け取った側に読めるようなかたちでコンピュータによって表示します。そうして送り手が書いた内容が受け手に伝わる、という仕掛けです。

仕掛けはともあれ、一般の郵便も電子メールも誰かが書いた文章をある特定の相手に届けるという本質は変わりませんね。

後の 5.1.7 でもっと具体的な利点などを紹介します。

### 5.1.3 ネットニュースって何？

電子メールが個人と個人間のメッセージ交換であったのに対して、ネットニュースは個人が大勢に対してメッセージをやり取りできるシステムです。

ネットニュースはあたかも掲示板に個人がメッセージを書くようなもので、大勢の人が（もしも興味があれば）そのメッセージを読み、そして自分の意見を再びそれぞれ大勢の人に見えるように書き込みます。こうすることによって、広く一般に向けて公開の議論が可能になるというわけです。

一般の掲示板は木の板にチョークか何かで書くのですが、ネットワーク上のニュースシステムでは、コンピュータを介する事によってメッセージをデータ化して行います。木の板の文字がデータかはともあれ、個人が書いたメッセージを多くの人が覗いて公開の返事を書くという本質は変わりませんね。

後の 5.1.8 でもっと具体的な利点などを紹介します。

### 5.1.4 WWWって何？

WWW では電子メールやネットニュースと違って、文字以外の絵や、音、動画などが簡単に扱えます。

そして、WWW はネットニュースと同じく掲示板のようなものです。但し、皆で共通の掲示板を読んだり書き込んだりするのではなく、誰かがその人専用で設けられた掲示板に自分で書き込むのです。その人が編集する雑誌のようなものだと考えてもいいかもしれません<sup>4</sup>。そこに詰め込まれる情報には適度なま

---

<sup>4</sup> 音や動画が含まれているのですから「飛び出す絵本」と呼ぶのが妥当かもしれません。

とまりがあり、それをページと呼んでいます。そのため、WWW の中身のことを「WWW ページ」などと呼んでいます<sup>5</sup>。

WWW には映像や音を扱える以外に、もう一つ特徴があります。「リンク」と呼ばれる、そこをマウスでクリックするだけで簡単に他のページを開くことができる仕掛けです。このリンクをたどる事によって、WWW ページを眺める人は、世界中のあちこちにある、いろんな人が作った WWW ページを簡単に眺めて回る事ができるのです。

つまり WWW ページは雑誌のようなもので、WWW 全体が大きな書庫のようなものだと思えば良いのでしょうか。そして書庫の中の雑誌のページはところどころリンクで結ばれているのです。

後の 5.1.9 でもっと具体的な利点などを紹介します。

### 5.1.5 京都産業大学のネットワーク

京都産業大学のコンピュータは、その多くが学内のネットワークに接続されています。計算機センターが管理しているコンピュータの多くも学内ネットワークに接続され、また上記のネットワークサービスが受けられます。

京都産業大学のネットワークは全世界を覆うネットワークに接続されています。この巨大なネットワークのことを インターネット と呼んでいます<sup>6</sup>。京都産業大学のコンピュータの多くは、つまりインターネットにつながれているということです。そして京都産業大学で行われているネットワークサービスの多くはこのインターネットで行われているものと同じものです。それ故に京都産業大学の電子メール、ネットニュース、WWW は、世界中の電子メール、ネットニュース、WWW と通じ合えるという訳です。

「電子メール」という呼び名は時々、インターネットで利用されている電子メールであるという意味で、「インターネットの電子メール」または「Internet mail」などと呼んだりもします。同様に「ネットニュース」は「Internet news」とも呼ばれます。WWW に関しては、現在のところインターネット以外の場には存在しないので、特に「インターネットの」と断る必要は余りありません。

あなたも京都産業大学内のコンピュータを介して世界中の人と電子メールを交換することができます。また、学内のコンピュータを介して世界中の人とネットニュースによって意見を交換することができます。WWW のページを作って、世界中の人に情報を提供することもできるのです。

### 5.1.6 インターネットとは？

ところでインターネットとは一体なんなのでしょう？よく「インターネットとはネットワークとネットワークを接続したもの」という説明を目にします。しかしこれは単にその構造を表現したものに過ぎません。そのような技術的な説明では無く、利用者にとっての価値を説明しなければ今はもう通用しないくらいインターネットは技術者以外の人のものになってきました。

ネットワークが人と人との情報交換の「場」であることは既に説明しました。即ちインターネットは世界じゅうを覆うくらいに地理的に拡大した、巨大なネットワークユーザ達の「場」なのです。そしてその「場」では、ネットワーク利用者は国境を越え、文化圏を越えて、対等に向き合うことができます。これは今までに無かったことです。そこからどんなに素晴らしいものごとが生み出されるか、想像してみてください。そしてその場に立ち合うばかりでなく、参加することだってできる自分の姿を想像してください。「インターネット」にこんなに多くの人ワクワクする、その理由の多くはそこにあるのです。インターネットとは、つまりそういうものなのです。

<sup>5</sup> よくインターネット関係の雑誌で「インターネットのホームページ」などと呼ばれていますが、これは余り妥当な呼び方ではありません。

<sup>6</sup> 「なんだか話が SF チックになってきましたね。」このガイドの最初の版、つまり 1994 年にそう書いたものですが、今となっては常識に近付いてきましたね。

## 用語と構造的な特徴

それでも少しは構造的、技術的な話をしないとイケません。インターネットの特徴として、主に以下のような点が挙げられます。

- インターネットには主体となる運営組織は具体的には存在しません。その代わりに多数の委員会の議論と実験によって運営されています。
- 中心となるコンピュータはどこにも存在しません。
- 世界中にある各組織の組織内ネットワークを接続して、協調して電子メールやネットニュースのサービスを提供しています。(各サービスについては以降にもう少し説明します。)

商用のいわゆるパソコンネットとはこれらの点で異なります。以下のようなことも知っておくといいでしょう。

- 国内ではインターネットに参加している組織は、大学などの教育機関、企業、国立の研究機関などで、'99年4月現在で約5万組織です。全世界だと、数百万組織は接続しているのでしょうか？
- 各組織のネットワーク(特にメールサービス)の管理者をポストマスタといいます。

これ以上の情報を得るためには、インターネット関係の雑誌やWWWページ等から自分で勉強するように努力してみてください。図書館にも置いてある各種のコンピュータ雑誌も推薦できます。

### 5.1.7 Internet mail サービスってどんなもの？

インターネットで行われている電子メールサービスを利用すると、世界中に散らばる100万もの組織の人達に対して手紙を出すことができます。この電子メールサービスは郵政省が行なっているメールサービス(郵便)に比べると以下の点で優れています。

- 速い  
たとえ相手が地球の裏側でも、大抵の場合(控え目に見積っても)数分間以内に相手の手元に手紙が届きます。
- いつでも使える  
基本的に24時間年中無休です。日曜日に休んだりはしません。
- 居ながらにして出せる  
京都産業大学のネットワークに接続されたコンピュータがありさえすれば、そこから全ての操作ができます。自宅から作業をする事もできます。
- 紙がたまらないで済む  
頻繁に手紙のやり取りをしていると、結構大きさのまちまちな保存しにくい紙がたまるものです。しかし電子メールでは手紙はコンピュータの中に残り場所を取りません。そして、消さない限りいつでも取り出して読み返せます。更に、計算機の中に入っている限りキーワードで検索可能です。紙をめくって斜め読みする必要はありません。

逆に、以下の点では劣っています。

- 書留郵便がない  
出したメールを必ず（何があっても）相手に届けるようなサービスは行なわれていません。メール配送の途中で、停電、機械の故障など何らかの理由によってメールが紛失する可能性があります。人生を左右するような大事なメッセージの交換は、決して電子メールだけに頼ってはいけません。
- 自分がネットワークに加入していないといけない  
しかもコンピュータを使わなければなりません。しかし最近ではコンピュータもずいぶん使いやすくなりました。
- 相手がネットワークに加入していないといけない  
しかもコンピュータを使っていてくれなければなりません。しかし最近では結構そういう人も増えていきますので問題ないのかもしれませんが。

電子メールは世の中の殆どの便利なものと同じ様に、決して万能ではありません。しかし、ある局面では他の何ものにも代えられない位役に立つことがあるのです。

### 5.1.8 Internet news サービスってどんなもの？

Internet news は NIFTY-serve など、商用のいわゆるパソコンネットの電子掲示板 (BBS) システムに似ています。つまりたくさんの人が掲示板に自分の言いたいことを書き込むというものです。この掲示板はまた多くの人に見られていますから、読んでいる人が書き込まれたことに反論したり、意見を添えたりします。

具体的には「私は今度行われる教育改革には反対だ。」「いや、私はこの点で賛成する。」「私も賛成だ。」と言った議論や、「先日若狭に釣りに行ってきました。今は鯛が好調です。」「私も行ってきました。少し沖合いに出るとハマチが来ます。餌は生き餌がいいようです。」「私も行きたいのですが、舟を紹介してください。」と言った趣味の話などが行われています。

また、何かについて困っているときに質問を書き込めば、誰か親切な人が解決法を教えてくれるかもしれません。そんな都合の良い話は無いです？いいえ、これは冗談ではありません。ネットワークの向こうには何千人もの非常に親切でお節介な人が暮らしているのです！<sup>7</sup>

要は多くの人が見ているところに書き込むわけですから、勢い情報交換の場となるのです。しかもこれは世界中の人が参加しています。つまり我々は海外の情報も居ながらにして読むことができるのです<sup>8</sup>。

非常にたくさんの方が参加して、いつも多くの書き込みがありますから興味のある話題ごとにグループ化されています。その中には、京都産業大学の内部向けに幾つかのグループが用意されています。外部の情報には興味がない人も、せめてここだけでも覗いてください。有用な情報の広報などはここでよく行われます。

### 5.1.9 WWW サービスってどんなもの？

WWW の利点については、ここであれこれ説明するより、テレビや雑誌を見る方がはやくかもしれません。そこではさまざまな会社や個人の、見た目も綺麗な WWW ページが紙面を飾っていたりします。きつと雑誌などには丁寧なその解説までしてくれている事でしょう。

でも少しはここにも書いておきましょう。WWW による情報提供には、次のような特徴があります。

<sup>7</sup> 嘘だと思ふならコンピュータ関係のネットニュースを少し覗いてみてください。

<sup>8</sup> もちろん海外のネットニュースは英語が殆どです。でも日本国内のネットニュースは日本語（漢字）で流れていますから、英語を読みたくない人もご安心ください。

- 速報性がある。
- 見たい人がわざわざ見に来ないと見られない。
- もっとよく見たい人のためにどんどん詳しい情報を置いておける。
- 滅多に見ることのなくなったような、過去の情報を残しておいてもコストがほとんどかからない。
- 視聴率(?) に関係なく、非常に多くの人に提供できる。
- 非常に安価に提供できる。
- 読者からの情報を集めたりすることもできる。

ところで WWW には、大企業が出している、自社製品の広告ページもあれば、学生が自分の趣味を紹介している極めて私的なページもあります。そのどちらが WWW の本流という訳ではなく、これら全てのもを対等に受け入れて混ぜ合わせたものが WWW の姿なのです。インターネットにつながれたコンピュータに WWW のページを作って置くだけで、あなたも自分の WWW ページを使って情報を発信することが、つまり世の中の人に何かを伝える事ができます。その可能性を大切にしてください。

#### 5.1.10 ネットワークでの暮らし方

ネットワークサービスと共にコンピュータを利用するのは非常に快適なものです。ですが、その環境で快適に暮らすにはある程度ルールを心得ておくことが重要です。ネットワークサービスは機械によって提供されていますが、相手をしているのは機械ではなく、人間であることをとかく忘れがちです。

1の「インターネット利用に関するガイドライン」などでも述べているように、京都産業大学のコンピュータ環境には一般社会と同じ様に規則、慣習、道徳があり、そして法律も適用されます。京都産業大学のネットワークはインターネットの一部でもあります。京都産業大学のコンピュータ利用環境よりもっと大きな利用環境であるインターネットにも勿論様々な規則、慣習があります。お互いに協調して暮らして行けるように心に留めておきましょう。

## 5.2 電子メール準備体操

### 5.2.1 Internet mail アドレスについて

インターネットでサービスされているメールサービス<sup>9</sup>でも一般的な郵便システムと同じ様に、宛先の住所を明記する必要があります。インターネットは世界中でサービスされているので、世界で一意に決まる<sup>10</sup>住所がメールサービスを受ける人それぞれに必要です。すなわちそれが住所であり、Internet mail サービスがアメリカ生まれであることからアドレスと呼びます。メールサービスのためのアドレスですから、メールアドレスというとその意味合いがもっとはっきりするでしょう。

現在のところ、メールサービスがコンピュータを介して行われているものであるため、メールアドレスはそれぞれのコンピュータの利用者に割り当てられます。逆に言えばメールサービスを利用するためには、メールサービスが利用可能なコンピュータの利用者とならなくてははいけません。

京都産業大学の幾つかのコンピュータは、Internet mail サービスが利用可能です。それらのコンピュータの利用者は Internet mail サービスを利用することができるでしょう。つまり Internet mail サービスを

<sup>9</sup> これ以降単にメールと言ったら電子メールを指します

<sup>10</sup> 一意に決まる、とは「間違いなくたった一つのものに特定できる」という意味です。

受けている世界中に数百万台（もっとかな？）存在するコンピュータのユーザそれぞれとメールを交換することができるという事になります。

京都産業大学のメールサービスが利用できるマシンにおけるメールアドレスは一般的には以下のような書式となっています。

foo@bar.kyoto-su.ac.jp

アドレスの表記そのものは西欧式になっており、右側に大きな区分、左側に小さな区分が書かれています。ピリオド(.)で区切られており、右側からそれぞれの区分の意味を以下に示します。

- jp            Internet mail アドレス最大の区分で、国を示しています。
- ac            jp 以下に存在する中区分であり、教学関係であることを示す。他に co が一般企業、or がその他の組織などとして定義されています。
- kyoto-su    京都産業大学を示しています。
- bar          ここには何が来るか一概には言えません。京都産業大学内で、既に決められている宛先に配送されることを示しています。大学のネットワーク管理者が決定した区分のようなものだと考えてください。
- @            これより左はユーザ ID であることを示しています。
- foo          foo ( 仮称 ) というユーザ ID 宛てに配送されることを示しています。

kyoto-su.ac.jp は、京都産業大学のネットワークを示すアドレスであり、世界中を覆う Internet で一意な名前です。ネットワーク環境の中で、このように階層付けされて一意に確保されているような名前をドメインなどと呼んだりします。kyoto-su.ac.jp より左の foo@bar は京都産業大学内で一意に保たれるように管理されています。あなたが電子メールを利用するときはあなたが利用者として登録されているコンピュータの管理者に、自分のメールアドレスがどのようなものかを確認することが重要です。

## 5.2.2  計算機センター運用のコンピュータのメールアドレス

ここでは計算機センターが運用しているコンピュータのメールアドレスに限定して説明します。ここで説明するルールは普遍的なものでは無いので、他のマシンのユーザのメールアドレスを類推する役には立たないでしょう。他のマシンのメールアドレスについては、それらのマシンの管理者に問い合わせる必要がある事に注意してください。

計算機センターが運用し、京都産業大学の教員、学生に提供しているコンピュータのうち、メールサービスが利用可能なのは以下のマシンです。ホスト名とはネットワーク上の各コンピュータの名前です。以降各マシンはホスト名で表現します。

機種名	ホスト名	
Enterprise3000	cc2000	計算機センター 1 階に設置の Sun 社製コンピュータ。
Ultra1	cssol000 ~ 043	2 号館 4 階 21 情報処理教室に設置の Sun 社製コンピュータ。
Netfinity3000	axt21001 ~ 46000	10 号館情報処理教室に設置の IBM 製コンピュータ。

以上のマシンでメールを利用する場合、メールアドレスは次のとおりとなります。

### メールアドレス

username@cc.kyoto-su.ac.jp

上記のうち@より左の username には各ユーザのユーザ ID を書きます。例えば産大太郎さんが taro というユーザ ID で cc 環境にユーザ登録されているとすると、メールを扱うためのメールアドレスは

となります。

ところで太郎さんは上記の他にもメールアドレスを持っているかもしれません。学校以外に契約してユーザ登録していて、そちらのメールアドレスを持っているときなどです。これは自然なことです。例えば太郎さんが二箇所に部屋を借りているような状態だと想像すれば良いでしょう。しかし太郎さんにメールを送る場合には、上記のうちのどれに送るのが妥当かは送る前に太郎さんに電話なり手紙なりで確認すべきです。つまり太郎さんは二箇所に部屋を借りていて、二つのメールボックスを持っているが、そのどちらを毎日チェックしているかは誰にも想像できないからです。これについては次の「相手のメールアドレス」にもっと詳しく書きます。

ところでよくよく考えてみると太郎さんは cc2000, cssol ~ マシン群及び axt ~ マシン群と合計すると数百台に及ぶ非常に多くのマシンに登録されていることとなりますね。しかしそんなにたくさんメールボックスがあっては面倒なので、これらのマシン群はメールボックスを共有しており、username@cc.kyoto-su.ac.jp 宛てに送られたメールは、この共用のメールボックスに配送されます。故に cc2000 及び cssol ~、axt ~ マシン群のいずれのコンピュータを用いてメールボックスをチェックしても、配送されてきたメールを読むことができます。どのコンピュータからメールボックスをチェックしたとしても、結果的には共通の、たった一つのメールボックスを見ている事になるからです。

### 5.2.3 相手のメールアドレス

いざメールを誰かに送ろうとした場合、相手のメールアドレスが必要になるでしょう。相手のアドレスを調べるには、その相手に聞くしか方法がありません。例えば産大太郎さんが上記の計算機センター管理のコンピュータ、cc2000 や cssol ~、axt ~ に hanako でユーザ登録されている産大花子さんにメールを出そうとした場合、hanako@cc.kyoto-su.ac.jp 宛てにいきなり出すのは好ましくないと云えます。コンピュータシステムは正直に cc2000 及び cssol ~、axt ~ コンピュータの為の hanako ユーザの共用メールボックスに太郎さんのメールを配送するでしょうが、花子さんが本当に cc2000 などのマシンでメールボックスをチェックしてくれるとは限らないからです。ひょっとしたら花子さんは電子メールなど全く使っていないかもしれません。ひょっとしたら、花子さんは他のコンピュータにも何らかの名前で登録されておりメールはもっぱらそこで利用しているかもしれませんね。

では太郎さんが花子さんにメールを送る場合はどうすれば良いのでしょうか？結論は「君に今後メールを送りたいのだけれど、いったいどこに送れば良いの？」<sup>11</sup>」と最初の一回目に（勿論メール以外の方法で）聞くことです。馬鹿馬鹿しいようですがこれは非常に重要なことです。

### 5.2.4 自分のメールアドレス

自分のメールアドレスが何であるかは自分がユーザ登録されているコンピュータの管理者に確認するのがいいでしょう。但し計算機センターが管理しているコンピュータについてのメールアドレスに関しては先に述べた通りです。

ある人にメールを送って欲しいと思った場合は、どこ宛てに送って欲しいか、すなわち自分が日常的にチェックしているメールアドレスを相手にはっきり通知することが大切です。これは先に書いた相手のメールアドレスをはっきり聞く、という事の裏返しです。同じく非常に重要なことです。

---

<sup>11</sup> 当然関西弁でも可

### 5.2.5 さあ、本番！

準備体操はこのくらいにしておきましょう。実際にメールを読んだり書いたりするのは現在ではコンピュータを操作するという事にほかなりません。即ち普遍的に「こうすればメールを読み、書く事ができる」という方法はなく、そのコンピュータ独自の操作方法を修得する事になります。

次に計算機センターが運用している cc 環境の UNIX コンピュータを利用してメールを読み書きする方法について説明します。

## 5.3 MHE : Mule による電子メールの読み書き

Mule という名前をここで初めて見た、もしくはほとんど覚えていない人はここまで読み飛ばした人ですね。早くメールを使いたい気持ちは分かりますが、ちょっと焦りすぎです。まずは 87 ページ、4.5 章を読んでください。

MHE<sup>12</sup> は Mule を利用して電子メールを読み書きする機能を提供します。Mule と共に働きますから、Mule の操作方法についてある程度理解していることを前提に説明します。

### 5.3.1 はじめに

ここでは以下の流れに従ってメールを扱う方法を説明します。

- メールを読む
- メールを書く
- 来たメールの返事を書く
- メールの整理

この流れのとおり、まずメールを読む方法を説明したいところなのですが、ちょっと問題があります。つまり恐らくあなたはまだ誰からもメールを送ってもらっていないので、メールを読む練習をするにも、読むべきメールが届いていないだろうということです。

そこでまず練習のために、自分自身にめがけて実験メールを送るコマンドを紹介します。mailself コマンドです。

```
cc2000(81)% mailself
cc2000(81)%
```

mailself コマンドは実行に数秒以上掛かります。うまく実行できた場合は、上記のように何もメッセージを表示せずに終了し、プロンプトが返ってくるでしょう。これで一通、新しいメールがあなた宛に届いているはず。次の節からは、そのメールを読むことで練習して行きましょう。

MHE の全ての操作は Mule 上で行います。さあ、mule コマンドで Mule を起動してください。

---

<sup>12</sup> 名前の由来は「Emacs front end to the MH mail system」から来ています。mh についてはここでは説明しません。man mh でマニュアルが用意されていますが、特に理解する必要はありません。Emacs (「いーまっくす」と読みます) とは Mule の古いバージョンです。

### 5.3.2 メールを読む

メールを読むためには、Mule が起動されている状態で M-x mh-rmail <Return>とします。<ESC> を1回ポンと押して x を押すと M-x と表示されますので続けて mh-rmail と入力し、<Return>です。

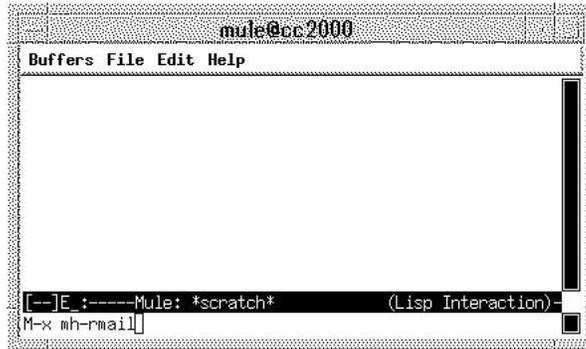


図 5.1 mh-rmail の起動

新規にメールが届いている場合はこれで以下のような画面表示となるでしょう。



図 5.2 メールの一覧表示

到着したメールが一覧として一通一行の形で表示されているでしょう。カーソルは一覧表示の左側にあるはずですが、ここでカーソルを上下させて、自分が読みたいメールの行の左にカーソルを移動させます。カーソルの上下は Mule 上でのファイルの編集の際のカーソルの上下と同じです。つまり上（一つ前の行）に移動したいときは上矢印（`↑`）もしくは C-p キー、下に移動したいときは下矢印（`↓`）もしくは C-n キーです。M-< や M-> で一番先頭や末尾の行への移動ができます。

自分が読みたいメールの行の左にカーソルを移動させて「. (ピリオド)」を押せば、そのメールの内容が Mule のウインドウを二分割して下半分に表示されると思います。

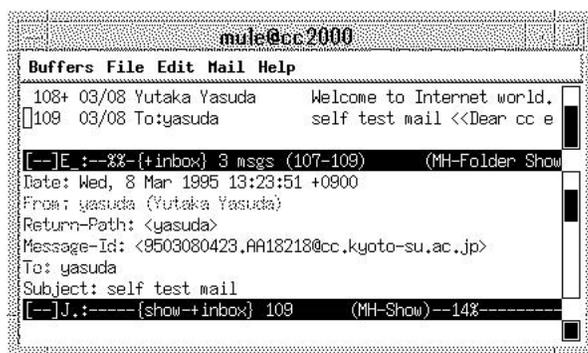


図 5.3 メールの内容の表示

一つのメールの行が長すぎて Mule のウインドウにおさまり切らないときは <Space>キーで一画面分スクロールさせる事ができます。スクロールして過ぎてしまった部分の内容を巻き戻してみたい場合は <Delete>キーです。

#### 用語説明

まず一覧に現れる情報について少し説明しておきます。以下に一覧表示の例を挙げます。

```
19+ 03/17 Yutaka Yasuda      Welcome to Internet world. <<Dear friends,
```

左から順番に項目別に説明して行きます。

19	順番に振られる番号です。番号が新しければ新しいほど昔の、古いメールです。
+	現在このメールに注目しているよという印です。
03/17	日付です。
Yutaka Yasuda	メールの送り元、つまり誰から来たかを示しています。 mailself コマンドで送り付けたテストメールであれば恐らくここは To: に続いてあなたのユーザ ID が来るでしょう。
Welcome to Internet world.	メールの表題です。
<<	ここから内容の一部だよという印です。
Dear friends,	恐らくまだ続きがあるはずですが、これがそのメールの本文の先頭部分です。表題だけでメールの内容の想像が付かないときなどは重宝します。

上記の例での Welcome to Internet world.、つまり表題はメールの世界では Subject と呼ばれ、本文とは少し区別して扱われます。つまりメールは主に「宛名」「送り元」「Subject」「本文」の4つの部分からなっているというわけです。

メールの一覧表示とはつまりこの主たる4つの部分のうち、宛名を除いた三つを並べているという事です。この一覧が並んでいる状態を、フォルダモードと呼んでいます。

さて、今度はメールの内容を読んでいる時に表示される部分の初めの方に、必ず付いてくる To: などに導かれた数行に注目してください。これらはメールのシステムが付けたもので、そのメールの補助的な情報が記録されています。一般の郵便で言うと表書きや消印の情報に相当します。この部分をメールのヘッダと呼んでいます。以下にメールヘッダの例を挙げます。

Date: Mon, 21 Mar 1994 12:53:31 +0900  
From: Tanaka Hideki <tanaka>  
Return-Path: <tanaka>  
Message-Id: <9501130246.AA07556@cc.kyoto-su.ac.jp>  
To: yasuda  
Subject: Re: NOMIKAI again  
Cc: tanaka, ryo, omatsu

以下に順番にそれぞれの行ごとに説明して行きます。

---

Date:	このメールが書かれた日付です。
From:	誰から送られてきたかを示しています。ここでは tanaka さんです。Tanaka Hideki は tanaka というユーザ ID の人のフルネームを示しています。
Return-Path:	返事を書いた場合の宛先になりますが、気にしないでください。
Message-Id:	これまた気にしないでください。
To:	誰宛に送られた手紙かを示しています。ここでは yasuda さん宛ですね。
Subject:	表題です。これはこのメールを書いた人が付けたものです。既に説明しましたね。
Cc:	Carbon Copy を意味しています。Carbon Copy とは複製を意味していて、このメールの複製を誰宛に同時に送ったかを示しています。ここでは tanaka, ryo, omatsu の 3 人に送っています。

---

#### 次のメールを読む

図 5.3 のようにメールの内容が表示されている状態で、(もしあったとして)次(もしくは一つ前)のメールを読むには幾つかやり方があります。

- 図 5.3 の状態のまま、カーソルを一つ下(もしくは一つ上)に移動して、そこでもう一度「.(ピリオド)」で内容を読む。
- 図 5.3 の状態のまま、n キーを押して次のメール(もしくは p キーを押して一つ前)の内容をすぐに表示させる。

#### MHE を終る、再起動する

メールを読み終って、普通の Mule の操作に戻りたいと思ったときは q キーを押します。これで MHE を起動する前、つまり図 5.1 の状態に戻ります。Mule を終りたい場合はいつも通りに C-x C-c です。再びメールを読みたいと思った時は単にもう一度(必要ならば)Mule を起動し、M-x mh-rmail<Return>とだけです。

以前読んでしまったメールを再び読み返す

MHE を再起動してまたメールを読みたいと思っても、mh-rmail はまだ一度も読んでいないメールの一覧を表示しようとしますから、前回メールを読んでから新しいメールが来ていなければ「読んでいないメールは一つも無いよ」と言ってメールを一行も表示しない時があります。あるいは以前に貰っているメールはもっとたくさんある筈なのに僅かしか表示しない時もあります。

そこで以前に読んでしまったメールをもう一度読み返したい場合は mh-rmail を実行してから M-r とします。以前に読んでしまったメールをさかのぼって表示してくれます。とりあえず最近の数通が一覧表示されます。それ以前のメールが見たい場合は C-p もしくは上矢印 ( ) キーなどでカーソルを上を移動させてさかのぼっていきましょう。状況によっては C-p しても何の変化もない場合があります。その場合でも実は画面のさらに上に古いメールが持ってこられている場合がありますから、確認のためとにかく C-p してみると良いでしょう。

### 5.3.3 メールを書く

さて、メールを読むことができるようになったら今度はメールを書いて送ってみましょう。一番良いのは誰か知人に相手になって貰うことですが、それができないようならまず自分自身宛に送って、その結果をチェックするのがよいでしょう。

新たにメールを書いて発信するには Mule が起動されている状態で、M-x mh-smail<Return>とするか、mh-rmail を実行してメールの一覧が表示されている図 5.2 の状態で s もしくは m とします。

するとまずエコーラインに To: と表示され、宛先を問うて来ます。ここでメールを送る相手のメールアドレス (5.2.3 で説明したことに注意してください) をタイプします。<Return>すると今度は Cc: と表示され、Carbon Copy が必要かどうかを聞いてきます。Carbon Copy とは複写のことで、そのメールの複写を控えとして送ることを指します。もしもあなたがそのメールの Carbon Copy をどこかに (例えば自分自身にでも) 送りたいのであれば Cc: の問いかけに対してメールアドレスをタイプすることで答えます。もしも Carbon Copy が必要無ければ単に<Return>してください。

To: や Cc: に書くアドレスは幾つでも並べて書けますので、複数の相手に同じメールを同時に送ることも可能です。並べて書くときはアドレスとアドレスの間に「, (カンマ)」で区切りを入れてやらなければならないことに注意してください。

今度は Subject: と表題を聞いてきますので、何かわかりやすい表題をタイプしてください。Subject には漢字やかなのは使わず、アルファベットと数字、記号程度で表現してください<sup>13</sup>。

以上の To:, Cc:, Subject: をタイプし終ると画面表示が変わり、手紙の内容を書くウインドウが用意されます。

<sup>13</sup> 最近 Subject にも漢字が利用できる場合がありますが、これは相手が Subject に漢字を適用できるシステムを持っているか、いないかに依存しますからそれが確認できない限りは漢字は使わない方が無難です。因みに現在の cc 環境の MHE は漢字の Subject には対応していません。

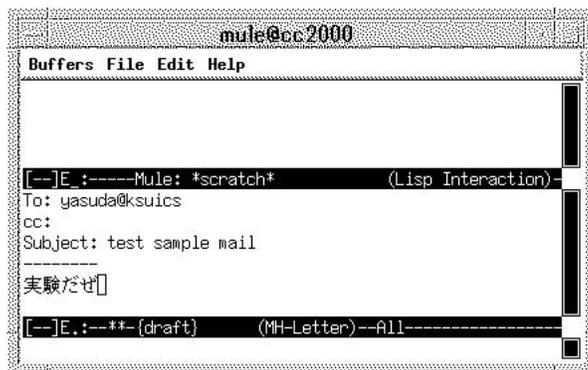


図 5.4 メールの内容を書く

この状態をレターモードと呼んでいます。To:, Cc:, Subject: のすぐ下に-----という行がありますが、これ以降に本文を書いてください。この---行は MHE システムが必要とするもので、削除してはいけません。この状態で普通に Mule によってファイルの内容を編集するときと同じ様にメールの本文を編集する作業ができます。Mule の操作に習熟すれば、メールの内容としてどこかのファイルの内容を取り込んだり、さまざまな応用が利くようになるでしょう。Mule の使い方、日本語の入力の仕方は 7 を参照してください。

そうやってメールの本文を書き終わったら、

```
C-c C-c
```

でメールが発信されます。反対にメールを書いている途中で、そのメールを出したくなくなった場合は C-c C-q とします。するとエコーラインに以下のような確認の為の質問をしますので、y と答えてください。

```
Kill draft message? (y or n)
```

これで、そのメールを破棄することができます。

#### 5.3.4 来たメールの返事を書く

来たメールに対する返事を書くには、返事を書こうとしているメールを図 5.3 のような状態で読んでいるときに a キーを押します。すると、そのメールに関係している人の誰に対しての返事を書くのかをエコーラインに以下のようなメッセージを表示して質問してきます。

```
Reply to whom:
```

この質問に対しては以下のいずれかで返事をします。

返答	意味
from	そのメールの差出人、つまり From: に書かれている人に返事を送る。
<Return>	from に同じ
to	そのメールの差出人、つまり From: に書かれている人を To: に、そのメールの To: に書かれている人を Cc: に指定する。
cc	そのメールの差出人、つまり From: に書かれている人を To: に、そのメールの To: 及び From: に書かれている人を Cc: に指定する。
all	そのメールの差出人、つまり From: に書かれている人を To: に、そのメールの To: 及び Cc: 及び From: に書かれている人を Cc: に指定する。

上記の表のいずれかによって返答すると図 5.4 に非常によく似た状態になり、返事のメールの本文が書けるようになります。(これもレターモードです。)

まだ誰からもメールを貰っていない時にこの返事を書く練習をするには、先ほど自分宛に送ったメールに自分で返事を書くのが良いでしょう。この場合は書かれた返事は再び自分宛に送られることとなります。このように自分が出したメールに返事を書く場合には Reply to whom: の問い合わせに対しては必ず to で答えるようにしてください。<Return>や、from で返答すると、返事を書く為の画面が正しく生成されず送信に失敗する時があります。

#### 電子メールでの返事の作法

普通の手紙とは違って、電子メールでは返事を書くときに前回の手紙の内容を引用することが簡単にできます。以下の例のように引用と返答とを交互に混ぜながら返事を書くこと判りやすいメールとなることでしょう。

田中さん> 先日は飲み会お疲れ様でした。  
いえいえ。ありがとうございました。

田中さん> そこで話しておりました原稿の件ですが、酒の上の事とは言え、冗談という  
田中さん> わけでもなく、本当にお願ひしたいと思います。  
ぎょぎょ。やはり本気だったんですね。( ^\_^; )  
ちょっと怖い気もしますが、了解しました。引き受けさせていただきます。

このような感じでしょうか。以下にこの種の引用を行なう手順を説明します。

返事を書こうとしているときに C-c C-y とすれば、そこに元となるメールの内容が引用されます。引用の際は自動的に各行初めの部分に「この行は引用だよ」ということを示すための記号を付けようとして、以下の質問をしてきます。

Complete cite attribution string: (default Hideki)

この default の右側には元のメールを書いた人の名前の一部が表示されていると思います。例では田中秀樹(たなかひでき)さんの Hideki が表示されています。ここでは自分の好みの引用記号をしましょう。漢字も使えますのでここでは 田中さん<Return>としてみましよう。単に <Return>とすると最初に表示されていた Hideki が採用されます。

[あ]Complete cite attribution string: (default >) 田中さん

すると以下のように聞いてきます。

登録しますか? (y or n)

ここで登録すると次回からこの人から来たメールに対する引用記号として「田中さん」を最初に表示します。y と答えると、以下のように聞かれる場合があります。

前の登録が消えますがいいですか? (y or n)

これは以前「田中さん」ではない別の記号で登録されていたのだけれど、それを「田中さん」に置き換えて良いかという確認です。これにも y と答えれば良いのです。

さて、次に引用元の記事を表示しているウインドウとは別のウインドウが開いて

```
>>>> On Wed, 08 Mar 1995 21:09:34 +0900, Environment Test Worker <worker> said\  
:
```

と表示されると同時にエコーラインに

```
Showing reference header 1.
```

と表示されます。これは引用を開始する部分の先頭に、どのような記述を含めるかという質問です。n キーを押すことで Showing reference header の数字が 0~6 まで変わり、ウインドウに表示されるスタイルも変化しているのが判るでしょう。気に入った表示が見つければそこで<Return>してください。すると返事を書こうとしているところに指定した引用記号を付けて元のメールの内容が現れるでしょう。但しそのままではメールヘッダも本文も全ての部分が引用されますので、かなり不要な部分も含まれている可能性があることに注意してください。

こうやって引用された元々のメールのうち、今回のあなたのメールでその返答となる部分以外の行を消去して、先に挙げた例のように引用と返答とを交互に混ぜながら返事を書くことと判りやすいメールとなることでしょう。

### 5.3.5 メールの整理

メールをしばらく利用していると、送られてきたメールがたくさん溜ってきて邪魔になります。まとまった種類ごとに振り分けたり、不要なメールは削除することができます。

削除するには図 5.2 の状態で不要になったメールの行の左にカーソルを移動し、ここで「.(ピリオド)」ならそのメールの内容が読める、という状況でピリオドの代わりに d をタイプします。するとメールの一覧表示の番号のすぐ右に D がマークとして表示されます。この段階は単に削除すべきメールにマークを付けているだけで、実際の削除はまだ行われていません。どんどん削除したいメールにマークを付け続けて行くことができます。マークを付け間違えた場合は、D マークの行にカーソルを移動して今度は u キーをタイプすれば D マークが消え、マークを外すことができます。

マークされたメールを実際に削除するには x キーです。問い合わせなどはなく、すぐに削除を実行します。メールをいくつか消すと番号が飛び飛びになってしまいましたが、これを振り直したければ M-p とします。

振り分けるには図 5.2 の状態で目的のメールの行の左にカーソルを移動し、ここで o をタイプします。

```
Destination folder? +
```

このようにエコーラインに表示されますので振り分けたいグループ名を入力して<Return>を押します<sup>14</sup>。するとメール番号の数字の右側に^が付きます。同様に振り分けたいメール全てに^マークを付けて、x キーを押すと振り分けが実行されます。

さて、振り分けたメールの読み方ですが、M-f とすると

```
Visit folder [+inbox]? +
```

とエコーラインに表示され、グループ名を聞いてきますので、先ほどメールを振り分けたグループ名を入力して<Return>を2回押すと表示されます。

### 5.3.6 メールの実体はどこに？

さて、前節で無事に消すことができたメールですが、実は完全に削除されたのではなく、一覧に現れなくしただけなのです。だから間違って重要なメールを消した時はまだ復活が可能です。

MHE は各ユーザのホームディレクトリのすぐ下に Mail というディレクトリを用意し、メールをそこに保存しています。普通に操作をしていけば、MHE は全て inbox というところにメールを溜め込むのですが、その実体は Mail ディレクトリの下にある inbox というディレクトリです。そこに一通のメールを一つのファイルとして保存しています。ファイル名には順番に付けられた番号が使われています。

具体的には Mail/inbox/1 などという名前でメールが残っているはずですが、自分で確かめてみると良いでしょう。以下のようにすると溜まっているメールのファイルが表示されます。メールを読む時に番号が付いていましたが、それと同じ数字のファイル名で1通1通が保存されています。

```
cc2000(87)% cd          (カレントディレクトリをホームディレクトリに移動)
cc2000(88)% cd Mail/inbox (カレントディレクトリを Mail/inbox に移動)
cc2000(89)% ls          (ファイル一覧の表示)
```

また、ここには #120 のように # から始まる名前のファイルがあるかもしれません。このファイルはあなたが不必要だと判断して D マークを付けて消したメールです。消した、と思っていたメールは実はまだ完全に消えた訳ではなかったのです。#120 というファイルは元120番のメールです<sup>15</sup>。もし必要なメールを消してしまっても、このファイル名を元に戻してあげることで復活できるのです。これを知っていれば安心ですね。

さて、安全のためすぐに消えなくなっているのはいいのですが、いわばゴミが溜まっているのですからきれいさっぱり捨ててしまいたいですね。消すのは rm コマンドでした。一つ一つ rm #120 のように消してもいいのですが、たくさんあると消すのに一苦労です。rm #\* とすれば、# から始まる全てのファイルを削除します。誤って rm \* としてしまうと、必要なメールも含めて本当に「全て」のファイルを消してしまうのでこれを実行する時には十分注意してください。前もって ls コマンドで確認すると安心です。

振り分けたメールは inbox ではなく、そのグループのディレクトリ名に残されています。たとえば freind というグループに振り分けていた場合、同様に

```
cc2000(87)% cd          (カレントディレクトリをホームディレクトリに移動)
cc2000(88)% cd Mail/friend (カレントディレクトリを Mail/friend に移動)
cc2000(89)% ls          (ファイル一覧の表示)
```

このようにすることで確認できます。

<sup>14</sup> 初めて使うグループ名を入力したときはその名前のフォルダを作成していいか、コンピュータが確認してきます。y と答えてください。

<sup>15</sup> メール番号は 119 ページの図 5.2 でそれぞれのメールの左についている数です。図 5.2 では 108 番と 109 番が表示されていますね。

```
cc2000(88)% ls #*
```

(ここで出てくるファイル名が要らないものばかりであることを確認)

```
cc2000(89)% rm #*
```

(これで先の ls で出たファイルを消す)

cc 環境ではここで説明したrm #\*を実行するコマンドを用意してあります。mailclean<sup>16</sup> とプロンプトから入力してメッセージに従ってください。

### 5.3.7 メールが来ているかどうか確認する

ずっと Mule を利用している人ならば、時々隙を見て M-x mh-rmail<Return>してメールを確認すれば良いのですが、メールが来ているかどうかを確認するためにわざわざ Mule を起動するのが面倒だという人のために、メールが来ているかどうかを簡単に確認する方法を紹介します。from コマンドです。from<Return>です。以下に例を示します。

```
cc2000(88)% from
```

```
From yasuda Tue Mar 22 18:51:30 1994
```

```
From yasuda Tue Mar 22 18:51:51 1994
```

```
cc2000(89)%
```

上記のように from コマンドは未読のメールがあれば、一通あたり一行で「誰からか」「いつ届いたか」を表示します。もしも未読のメールが一つもなければ from コマンドは何も表示せずに以下のように終了します。

```
cc2000(88)% from
```

```
cc2000(89)%
```

from コマンドで未読メールが見つかった場合だけ、Mule を起動して MHE で読めば良い、という事です。

### 5.3.8 メールの印刷

せっかく電子メールを使っているのですから、不必要な紙への印刷は避けるべきです。しかしどうしても印刷したい時があるかもしれません。その場合は 95 ページ、4.6 で紹介した lpr コマンドを使います。

例えば、印刷したいメールの番号が 110 で、それを ccpr01 に印刷したいなら次のようにします。

```
cc2000(88)% cd Mail/inbox
```

```
cc2000(89)% lpr -Pccpr01 110
```

### 5.3.9 トラブルからの脱出

なんだか mule が変なんです。

いろいろな状況が予想されますが、C-g、C-g、C-1 と順に実行すると直るかもしれません。

---

<sup>16</sup> 一般的なコマンドではありません。

MHE を起動しようとするとき、`.mh-profile` がない、というエラーメッセージが表示されるんです

Cannot find MH profile /NF/home/..(略)../.mh\_profile というメッセージがエコーラインに表示され、MHE が起動できない場合があるかもしれません。このような場合は一度 Mule を終了してから、おまじないとして `inc` コマンドを一度だけ実行してください。何か問い合わせをしてくるかもしれませんが、その時は `y` と答えてください。それでもう一度 Mule を起動し、それから MHE を起動してみてください。

メールが来ているのに読もうとすると「No current message.」と出て読めません

長期に渡ってメールを利用すると、メールの番号がどんどん増えていきます。この数字が 10000 を超えてしまうと正しくメールを表示できなくなります。M-x mh-rmail の後、M-p で番号を振り直してください。

### 5.3.10 MHE もっともっと

ここに紹介したのは MHE の機能のうちの基本的なものだけです。MHE にはもっと様々な機能がありますが、それについてはここでは説明しません。附録に参考文献を挙げておきますので、それらを参照してください。

### 5.3.11 まとめ

メールを読み書きする時のコマンドをまとめておきます。カーソルの移動に関しては含みませんので他を参照してください。

M-x mh-rmail		mule をメールが読める状態にする
メールを読む時	.	カーソルの位置のメールを読む
	<Space>	メールの続きを表示する
	<Delete>	メールの内容をまき戻して表示する
	q	メールを読み終わる
	a	カーソルの位置のメールに返事を書く
	d	カーソルの位置のメールに消す印を付ける
	o	カーソルの位置のメールを振り分ける印をつける
	x	d や o で印を付けたメールの削除及び振り分けを実行する
	M-p	番号を 1 から順に振り直す
M-r	過去に受け取ったメールも表示する	
M-f	振り分けたメールを表示する	
M-x mh-smail		mule をメールを送る状態にする
メールを書く時	C-c C-c	メールを送信する
	C-c C-q	メールを送るのを止める
	C-c C-y	(返事を書いている状態で) 元のメールの文を引用する

### 5.3.12 signature

signature<sup>17</sup> とはメールの最後に付ける署名です。署名というと固いですが、誰の書いた文章かをわかりやすくするものだと思います。シンプルに

<sup>17</sup> しぐねちゃーと読んでください。

としてもいいですし、文字で絵を描いてみても楽しいですね。

```
( )
)~
"" "" ^ "" "" "" "" ""
京都産業大学
計算機センター事務室
尾崎 孝治
E-Mail ozaki@cc.kyoto-su.ac.jp
```

signature はあなたの顔みたいなものです。個性的で印象に残るものにしましょう。また、京産大ではお互いの顔が見えるような、アットホームなネットワーク環境作りを目指しています。その一環として自分の名前を名乗ろう、ということになっています。ご賛同頂けるなら学科や学年と名前を含めるようにしてください。学年まで必要なのか？と疑問に思う方もいるかもしれませんが。「京都産業大学 理学部 計算機科学科 尾崎 孝治」ではいけないのか？ということです。でも、もしかすると尾崎さんは理学部の教員かもしれません。それに気づかず気軽にメールに返事を書いてしまうと失礼な内容になってしまうかもしれません。そのような事故は防ぎましょう。学年を名乗るだけで簡単に防げるのですから<sup>18</sup>。

しかし、signature の行数があまり多くなると記事自体が読みにくくなってしまいます、少ない行数に情報をきれいにまとめるところが腕の見せ所です。目安としては 4 行以下がいいでしょう。

もちろん無理に signature を付ける必要はありません。思想的、またはいろいろな理由で signature をつけたくないと思う人もいるでしょう。けれどインターネットの社会では匿名で発言することはマナー違反です。ですからその場合、一番最初に学科、学年、本名を名乗るようにするといいでしょう。

また、自動的に signature を付ける機能も用意されていますが、使うコンピュータによってはうまく動かないのでお勧めしません。お勧めは手動で差込む方法です。適当な名前、例えば sig という名前のファイルに signature を作って保存してください。そして必要な時に C-x i とすると Mule の機能で、カーソルの位置に別ファイルの内容を挿入できる状態になります。そこでファイル名を入力し、<Return>とすれば確実に signature を付けることができますし、毎回 signature を書く手間が省けます。

### 5.3.13 メールを書くときの注意

ここでは操作方法では無く、メールの中身の書き方について説明します。

初めてメールを出すときに

始めのうちはいきなり学外や海外にメールを出さずに、学内でメールの練習をしてから送ってください。練習の相手が見つからない場合は自分宛に送ってみてください。

あまり大きなファイルを送らないこと

メールによってファイルを転送することもできますが、

- どうしてもメールによってファイル転送する場合は目安として 50 キロバイト<sup>19</sup> /メール以下にしましょう。あまり大きなファイルを送るのは、参加組織に迷惑がかかります。
- どうしてもメールによって大きなファイルを送らなければならないなら、1 メガバイトくらいまでなら分割して送っても大丈夫かもしれません。それ以上ならフロッピーやテープで送りましょう。

<sup>18</sup> 勿論学年を名乗らなくても「教員」「学生」としてもいいですが、あんまり格好よくないです。学年を名乗る方が自然と思いませんか？

<sup>19</sup> バイトというのはコンピュータ上の資源の大きさを表す単位です。50 キロバイトと言えば漢字にして 25000 字、つまり原稿用紙 50 ページ強辺りとなります

そのメールは相手を読めるものですか？

現在のメールがコンピュータによって処理されている限り、自分が利用しているコンピュータによって処理できた文書が、相手を利用しているコンピュータによって復元できるとは限らない事に注意しましょう。

- 漢字を含むメールが必ず相手に読めるとは限りません。相手はひょっとしたら漢字が表示できないコンピュータを使ってメールを読んでいるかもしれません。相手が漢字を読めるかどうか、まず最初に確認するのがよいでしょう。
- 基本的にインターネットは JIS コード漢字を利用していますが、あなたがインターネットを利用する窓口になるコンピュータによって漢字コードはまちまちです。たとえば計算機センターが管理している cc2000, cssol ~, axt ~ に限っては標準の漢字コードとして EUC コードを採用していますが、メールは JIS コードに変換されて送られるように設定されています。どの漢字コードを適用すれば良いのか判らない場合はホストコンピュータの管理者に確認してください。
- Subject に漢字は使わないでください。
- 使ってはならない文字として、半角カナ（カナ・キーを押して入力したもの）やメーカー独自の文字（(株),I,II,III,IV,V 等の文字が 1 文字で表されているもの、丸の中に文字が書いてあるものなど）があります。また、使わない方がよい文字としては、罫線素片等があります。

#### メールの信頼性について

送ったメールは相手に確実に届くことも、内容の完全な秘匿性も保証されていません。到着の確実性が要求される場合には、相手にそのメールが届いたら折り返し届いた旨をメールしてもらおうように頼みましょう。ある程度の秘匿性が必要ならば暗号化するかまたは直接手渡ししてください。就職活動のメールなど、人生を左右するようなものならなおさらです。

#### 一行の長さについて

メール配送を行うコンピュータのシステム自体は一行 255 文字までサポートしています。しかし、そのメールの読み易さや返事を書く時に内容を参照することも考えて、漢字で約 35 文字以下、アルファベット（半角文字）で約 70 文字以下にするのが良いでしょう。

#### 海外へのメールについて

国内以上にアドレスや送るメールの大きさの注意が必要です。また、海外のネットワークにはそれぞれのネットでの取り決めがあるので、それに従うようにしましょう。

メールアドレスに工夫が必要な場合もあります。一般的には宛先のアドレスは user@aaa.bbb.ccc のように .jp で終らないものの、国内と同様の書式です。（user には相手のユーザ ID を、aaa,bbb,ccc 等には相手のアドレスを入れます）

#### 著作権および責任の所在

書いたメールに関する責任は、書いた人の属する組織にあるのではなく、書いた人自身にあります。また、著作権はそのメールを書いた人にあります。

## 違法行為の禁止

例えば、メールでのソフトの違法な流通などをやってはいけません。

## いたずら

いたずら電話のようにいたずらメールを送ることは当然してはいけないのですが、気づかないうちにいたずらメールに相当するメールを送ってしまうことがあります。例えば、このようなメールを受け取ったとき、あなたはどのように対処しますか？

「知り合いの女の子が事故に遭い入院しています。彼女はとてもまれな血液型で、病院のストックが足りず至急一致する血液型を持つ人を捜しています。どうかご協力をお願いします。また、残念ながらあなたの血液型が該当しなくても、このメールをできるだけたくさんの人に転送していただくようお願いします。……（以下省略）」

「これは大変だ。私が知っている全ての人に協力を頼んでみよう！」こう思ったあなたは心やさしい素晴らしい人だと思います。でも、これが予想もしない結果を招くのです。

いわゆる不幸の手紙と呼ばれるものをご存知でしょう。人の弱みにつけ込んでどんどんその規模を広げていくものです。ここで、先ほどのメールも内容的には天と地ほど違いますが、システム的には不幸の手紙と同じような効果を持っていることに気づくでしょう。結果的にネットワーク上を非常にたくさんの電子メールが飛び交うことになるのです<sup>20</sup>。また、人の良心に訴えかけていることと、電子メールは普通郵便と違い簡単に転送が可能なることにより、不幸の手紙より規模の拡大が速く、たちが悪いものになっています。この例にあげたメールの事件は実際に起こったことなのですが、有効期限が記されていないためいつまでもメールの転送が続き、収束するまでに大変な時間と（もう必要ないという情報を伝える）労力を要しました。

このようなことに電子メールを使用するときは十分な注意が必要です。通常、不特定多数に呼びかけるのならばネットニュースの方が適しています。この例のような場合はネットニュースを使うべきです。

先ほどの例は純粋な目的により起こった事件でしたが、いたずら目的で作り話をでっちあげ、ネットワークに混乱を与えようとする人もいます。そんないたずらは看破して、メールの転送はしないようにしましょう。参考までにもう一例あげておきます。

「Subject に Good Times とかかれた電子メールが届いたら、それはコンピュータウイルスに犯されていますので読まずに捨ててください。この情報をあなたの周りの人に教えてあげてください。」

どうですか？「よし！今すぐ教えてあげよう！」ではいけませんよ。そうそう。お金もうけをしませんか、という内容のものもあります。ほとんどが詐欺ですので注意してください。

<sup>20</sup> そのどんどんつながっていく様よりチェーンメールとも呼ばれます。

## 5.4 GNUS : Muleによるネットニュースの読み書き

GNUS は Mule を利用してネットニュースを読み書きする機能を提供します。

Mule と共に働きますから、Mule の操作方法についてある程度理解していることを前提に説明します。

### 5.4.1 用語説明

Internet news は生まれが USENET と呼ばれる大学間で始まった実験ネットワークです。その歴史的な経緯でネットニュースシステム上で用いられる用語が少し一般のパソコンネットの電子掲示板システムとは違ってきます。以下に列挙しておきましょう。

- ネットニュースは非常にたくさんの人が参加して、いつも多くの書き込みがありますから興味のある話題ごとにグループ化されています。これをニュースグループと呼んでいます。(「NG (エヌジー)」とも呼びます。)これはパソコンネットでは SIG もしくはボード、フォーラムなどと呼んでいるものに相当します。
- ネットニュースにユーザが書き込んだメッセージを記事 (アーティクル) と呼んでいます。
- 記事を書き込むことを投稿 (ポスト) と呼んでいます。

### 5.4.2 はじめに

ここでは以下の流れにしたがってネットニュースを扱う方法を説明します。

- GNUS の起動
- ネットニュースの記事を読む
- ネットニュースグループの選択
- ネットニュースの記事を投稿する
- ネットニュースの記事にフォローする
- 投稿した記事のキャンセル
- ネットニュースの記事にメールで返事をする
- 古い記事を読み返す
- ネットニュースの記事の保存

GNUS の全ての操作は Mule 上で行います。さあ、mule コマンドで Mule を起動してください。

### 5.4.3 GNUS の起動

ネットニュースを読むために、まず GNUS を起動します。Mule が起動されている状態で M-x gnu<Return> とします。

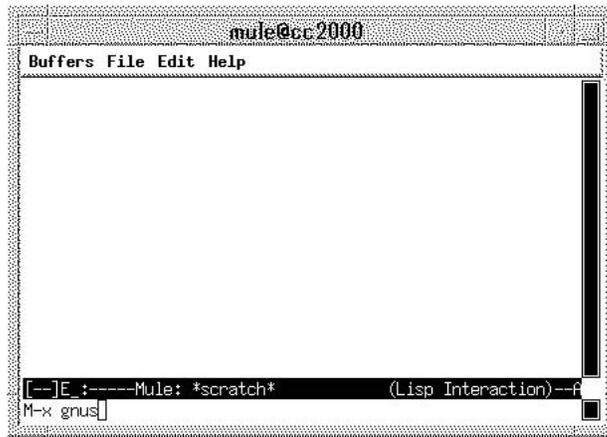


図 5.5 GNUS の起動

最初の GNUS の起動には 1 分以上かかる場合があります。これは全ての情報をゼロから構築しているためで、次回からはもう少し短い時間で起動できるようになるでしょう。以下のような状態になるまで、しばらくお待ちください。



図 5.6 ニュースグループ一覧

#### 5.4.4 記事を読む

図 5.6 のように、一つのニュースグループが一行で表わされ、一覧として表示されているでしょう。カーソルは一覧表示のニュースグループの名前の左側にあるはずで、ここでカーソルを上下させて、自分が読みたいニュースグループの行にカーソルを移動させます。カーソルの上下は Mule 上でのファイルの編集の際のカーソルの上下と同じです。つまり上（一つ前の行）に移動したいときは上矢印（`↑`）もしくは C-p キー、下に移動したいときは下矢印（`↓`）もしくは C-n キーです。M-< や M-> で一番先頭や末尾の行への移動ができます。

このニュースグループの一覧が表示されている状態を「グループモード」と呼んでいます。

おそらく今は sandai. で始まる産業大学ローカルニュースグループだけが一覧に表示されるはずで

す。'98.3 現在のローカルニュースグループの一覧を載せておきます。

内容	ニュースグループ名	期限
簡単な、もしくは一般的な質問	sandai.question	二カ月
娯楽及び趣味に関する話題	sandai.rec	二カ月
その他の、もしくは一般的な話題	sandai.misc	二カ月
下らない話題	sandai.junk	二週間
お知らせ、速報（一週間以内有効）	sandai.news.week	一週間
お知らせ、速報（一カ月以内有効）	sandai.news.month	一カ月
産業大学作品展示会場	sandai.gallery	
実験用ニュースグループ。貴方自身の練習用に	sandai.test	一週間
コンピュータに関する話題	sandai.comp	
学内コンピュータ環境に関するお知らせ	sandai.comp.announce	
コンピュータに関する管理者向けの話題	sandai.comp.admin	
授業に関する話題	sandai.classes	
授業に関する話題 (特に??クラスについて)	sandai.classes.??	
産業大学のネットニュース購読者全員へのお知らせ	sandai.announce	
産業大学の一般的な話題	sandai.general	
計算機センターからのお知らせ	sandai.center.announce	
計算機センターへの一般的な話題	sandai.center.general	

各ニュースグループに投稿された記事は、上記の期限を過ぎると自動的に消去されます。期限の書かれていないニュースグループの記事はそのまま残り続けます。この期限の設定を含めてニュースグループの設定は流動的なものです。最新の情報が知りたい場合は 4.7 で紹介した ccinfo コマンドを利用するか、sandai.question に質問を投稿するなどしてください。

さて、グループモードで、自分が読みたいニュースグループの行にカーソルを移動させて<Space>キーを押せば、そのニュースグループの記事の表題一覧と、そのなかの記事の内容を表示できる以下のような状態へと移行します。



図 5.7 記事の内容表示

この状態では Mule のウィンドウが二分割され、上半分に先ほど選んだニュースグループの記事の表題の一覧が、下半分にそのニュースグループの先頭の記事の内容が表示されています。下半分に表示されて

いる記事の内容が長すぎて Mule のウインドウに収まり切らないときは<Space>キーで一画面分スクロールさせることができます。スクロールして過ぎてしまった部分の内容を巻き戻してみたい場合は<Delete>キーです。

#### 次の記事を読む

今度は、ある特定のニュースグループの中の、一つの記事が一行で表わされ、一覧として表示されている訳です。カーソルは一覧表示の番号の右にあるはずで、ここでカーソルを上下させて、自分が読みたいニュースグループの行にカーソルを移動させます。カーソルの上下は Mule 上でのファイルの編集の際のカーソルの上下と同じです。つまり上（一つ前の行）に移動したいときは上矢印（`↑`）もしくは `C-p` キー、下に移動したいときは下矢印（`↓`）もしくは `C-n` キーです。 `M-<` や `M->` で一番先頭や末尾の行への移動ができます。

自分が読みたい記事の行にカーソルを移動させて<Space>キーを押せば、記事の内容が Mule のウインドウの下半分に表示されます。

また、記事の内容を読んでいるときに<Space>キーを押し続けて行けば、一つの記事を読み終った段階で自動的に次の記事に移動します。記事を読んでいる途中で `n` キーを押せば、今読んでいる記事の次の未読記事を表示します。 `p` キーを押せば、今読んでいる記事の前の未読記事を表示します。

この、あるニュースグループを選択して、記事を次々と読んで行ける状態を「記事モード」もしくは「アティクルモード」と呼んでいます。

`q` キーで記事モードから抜けてグループモードへ戻ります。図 5.6 のような表示になってグループモードに戻った後は、また違うニュースグループを選んでその記事を読むことができます。

### 5.4.5 GNUS を終了する、再起動する

記事を読み終って、普通の Mule の操作に戻りたいと思ったときはグループモードで `q` キーを押します。すると、エコーラインに以下のような表示が現れ本当に GNUS を終了するかどうか聞いてきます。

```
Are you sure you want to quit reading news? (y or n)
```

ここで `y` キーを押してやると GNUS を起動する前、つまり図 5.5 の状態に戻ります。Mule を終りたい場合はいつも通りに `C-x C-c` です。

再びネットニュースを読みたいと思ったときは単に `M-x gnus<Return>` とするだけです。

### 5.4.6 ニュースグループを選ぶ

#### 読みたくないニュースグループを一覧表示から外す

グループモードでは、何も指示しなければ既に示したように `sandai` ニュースグループを全て表示します。その中にはあなたにとってほとんど意味のないニュースグループもあるでしょう。もう読まないと決めてしまったニュースグループはこの一覧に表示させないようにすることができます。図 5.6 のようなグループモードでカーソルを読みたくないニュースグループの行に移動し、そこで `u` キーを押します。するとその行の左端に `U` マークが付きます。 `U` マークを付け間違ってしまった時は、もう一度その行で `u` キーを押すことによって外すことができます。幾つか `U` マークを付けていたら `l` (英小文字の `L`) キーを押してみてください。今まで `U` マークがついていた行が表示されなくなりましたね。

これで次から GNUS を使ってネットニュースを読む時に、全く読まないニュースグループの一覧が出てこなくなって随分楽になるというわけです。ちなみに u は Unsubscribed の略で「購読しない」ということを意味しています。

一覧に出てこなかったニュースグループを一覧に加える

cc 環境ではあらかじめ多くのニュースグループが Unsubscribed されています。最初の利用であれば sandai. で始まる産業大学ローカルニュースグループだけが一覧に表示されるはずですが。

しかし挙げたものは cc 環境で利用できるニュースグループのほんの一部に過ぎません。もっと多くのネットニュースを読みたい!と思った人は一度 Unsubscribed されているニュースグループを全て表示させてみるのが良いでしょう。グループモードで L (今度は英大文字の L) キーを押してください。(少し時間がかかります。)一覧に表示されている限り普通の操作でそのニュースグループの記事を読むことができます。

全部で 6000 以上のニュースグループがありますので最初は何を読んでいいのかわかるでしょう。そのうち fj. と tnn. と japan. で始まるニュースグループが日本語で書かれている主なものです。その他のほとんどのニュースグループは英語で書かれています。日本とアメリカ以外の各国のネットニュースを気が付いたところだけ並べてみました。参考にしてください。

can. uw. カナダ	de. zer. ドイツ	dk. デンマーク
fr. フランス	il. イスラエル	no. ノルウェイ
nz. ニュージーランド	swnet. スウェーデン	uk. イギリス
za. 南アフリカ共和国		

ただこのままでは次に GNUS を起動した時にはまた一覧から外れてしまいますから、目的のニュースグループを一覧に表示させた状態で、そのニュースグループについている U マークを再度 u キーによって外してやれば良いのです。その効果はグループモードで l (今度は英小文字の L) キーを押すことで確認できますね。

Unsubscribed されているニュースグループの名前があらかじめ判っている場合は、もっと簡単に一覧に表示させることができます。グループモードで j キーを押すと以下のようにニュースグループ名を聞いてきます。

Newsgroup:

ここで目的のニュースグループの名前をタイプして<Return>すると、それがニュースグループ一覧の一番上の行に表示されます。あとは普通の操作でそのニュースグループの記事を読むことができます。次回の GNUS の起動後に再び一覧に表示させておきたい場合はそのニュースグループについている U マークを再度 u キーによって外してやる事を忘れずに。

#### 5.4.7 記事を投稿する(けどちょっと待てよ)

せっかくネットニュースシステムが稼働しているのです。ただ読むだけで全く投稿しないのも面白くありません。ここは一つ何か書き込んでみましょう。

## 投稿する前にちょっと考えること

記事を投稿するのは GNUS で勿論可能なのですが、ネットニュースを読みはじめて間もなくの頃とはとにかく読むのに徹して投稿は控えた方がいいと思います。これはネットニュースというものをよく理解しない内に「的外れな」「失礼な」もしくは「迷惑な」記事を投稿する危険があるからです。

ちょっと否定的に書きましたが基本的にはネットニュースも含めてネットワークサービスは参加しないと意味がありません。人の書いたものを読むだけでは面白さも半減です。あまり遠慮しすぎる必要もありません。投稿そのものには賛成します。ネットニュースを盛り上げるためにもぜひ投稿してください。

但しあなたがまだ投稿する事に慣れていない場合は、まず最初に sandai.test ニュースグループに投稿してみましょう。ここは投稿の実験や、練習の為のニュースグループですから操作に失敗しておかしなメッセージを投稿してしまっても誰の迷惑にもなりません。とりあえずここに自己紹介のメッセージでも投稿してみてもどうでしょうか。また、投稿はしばらくの間 sandai で始まるニュースグループに限定するのがお勧めです。つまり fj や comp など始まるインターネットを経由して外部に流れ出てしまうニュースグループには投稿しない方がいいかもしれません。そういうところで投稿に失敗したら、最悪の場合失敗した記事が世界じゅうを駆け巡ることになってしまい、大変迷惑です。しばらく投稿の操作や作法に慣れてからの方がいいでしょう。

何事も順を追ってやりましょう、という事です。

## さあ投稿してみましょう

さてもう前置きは良いでしょう。何か記事を投稿してみましょう。記事の投稿には二つのケースが考えられます。

1. 全く新しい話題を投稿する
2. 人の記事に対する意見やコメントを投稿する。

ネットニュースをいくらか読んでいるうちに、この二種類の記事に遭遇すると思います。誰かが質問の記事を投稿し、他の誰かがそれに答えたりしているでしょう<sup>21</sup>。

まず先に 1. のケースについて説明します。次に 2. のケースについて説明します。2. のケースのように人の記事に対して意見を加えて投稿することをフォローすると呼んでいます。

### 5.4.8 新しい記事を投稿する

新しい記事を投稿するにはグループモードもしくは記事モードで a キーを押します。すると以下のようなメッセージをエコーラインに表示して確認をします y で答えます。

```
Are you sure you want to post to all of USENET? (y or n)
```

もしグループモードで a キーを押した場合は、以下のようなメッセージをエコーラインに表示してどのニュースグループに投稿するのか聞いてきます。記事モードで a キーを押した場合は、キーを押した時のニュースグループが投稿先のニュースグループとなりますので、聞いてくることはありません。

Newsgroup:

---

<sup>21</sup> え？見たことがない？それは余りに経験が少なすぎます。もうちょっとネットニュースを読み込んでみてから投稿してはいかがですか？

これにはポストする記事のニュースグループをタイプします。するともう一度同じように Newsgroup: と聞いてきますので、単に<Return>とします<sup>22</sup>。

すると今度は以下のようなメッセージをエコーラインに表示して投稿する記事の表題を聞いてきます。

Subject:

これには何かわかりやすい表題をタイプしてください。Subject には漢字やかなはつかわず、アルファベットと数字程度で表現してください<sup>23</sup>。

すると今度は以下のようなメッセージをエコーラインに表示して投稿する記事の配布範囲を聞いてきます。

Distribution: local

この Distribution は、投稿する記事の配布範囲を示しています。以下のいずれかを答えるようにします。

配布範囲	ニュースグループ	意味
local	sandai.general など	京都産業大学内のローカルニュースグループ用。京都産業大学以外に配布されることはない。
world	fj.jokes , comp.sys.sun など	国際ニュースグループ用。comp, sci, soc など世界じゅうに配布される。

あらかじめ local と表示されていますから、local で良い場合はそのまま<Return>、変更したい場合は local の後ろにあるカーソルを<Delete>キーで戻して world とタイプします。ここまで答えると Mule のウィンドウの状態が変わって、記事の内容を編集できるようになります。



図 5.8 記事を書く

ここで記事を書くわけです。Newsgroups:, Subject:, Distribution: のすぐ下に

--text follows this line--

<sup>22</sup> ここで複数回聞いてくるのはクロスポストという操作方法の為なのですが、ここでは説明しません。

<sup>23</sup> 最近 Subject にも漢字が利用できる場合がありますが、これは相手が Subject に漢字を適用できるシステムを持っているか、いないかに依存しますからそれが確認できない限りは漢字は使わない方が無難です。因みに現在の cc 環境の GNUS は漢字の Subject には対応していません。

という行がありますが、これ以降に本文を書いてください。この行は GNUS システムが必要とするもので、削除してはいけません。この状態で普通に Mule によってファイルの内容を編集するときと同じ様に ネットニュースの本文を編集する作業ができます。Mule の操作に習熟すれば、記事の内容としてどこかのファイルの内容を取り込んだり、さまざまな応用が利くようになるでしょう。

そうやって記事の本文が書き終わったら、C-c C-c で記事が発信されます。反対に記事を書いている途中で、その記事を投稿する気がなくなった場合は、C-c C-k とすれば、その記事を破棄することができます。

#### 5.4.9 記事にフォローする

誰かが書いた記事を読んでいて、その記事に対して意見を添えたり、質問をしたり、疑問に答えたりするような記事を投稿する事をフォローと呼んでいます。フォローを行うには、その元となる記事を読んでいる状態で F キーを押します。

すると、単に新たな記事を書いて投稿するときと同じ様に以下のようなメッセージをエコーラインに表示し確認してきます。

```
Are you sure you want to followup to all of USENET? (y or n)
```

これにも y と答えてやります。すると元となる記事の内容を引用するための記号を付けようとして、以下の質問をしてきます。

```
Complete cite attribution string: (default Hideki)
```

このの部分には元の記事を書いた人の名前的一部分が表示されていると思います。例では田中秀樹(たなかひでき)さんの Hideki が表示されています。ここでは自分の好みの引用記号をしましょう。漢字も使えますのでここでは 田中さん<Return>としてみましよう。単に <Return>とすると最初に表示されていた Hideki が採用されます。

```
[あ]Complete cite attribution string: (default >) 田中さん
```

すると以下のように聞いてきます。

```
登録しますか? (y or n)
```

ここで登録すると次回からこの人から来た記事に対する引用記号として「田中さん」を最初に表示します。y と答えると、以下のように聞かれる場合があります。

```
前の登録が消えませんがいいですか? (y or n)
```

これは以前「田中さん」ではない別の記号で登録されていたのだけれど、それを「田中さん」に置き換えて良いかという確認です。これにも y と答えれば良いのです。

さて、次に引用元の記事を表示しているウィンドウとは別のウィンドウが開いて

```
>>>> On Wed, 08 Mar 1995 21:09:34 +0900, Environment Test Worker <worker> said\  
:
```

と表示されると同時にエコーラインに

```
Showing reference header 1.
```

と表示されます。これは引用を開始する部分の先頭に、どのような記述を含めるかという質問です。n キーを押すことで Showing reference header の数字が 0~6 まで変わり、ウインドウに表示されるスタイルも変化しているのが判るでしょう。気に入った表示が見つければそこで<Return>してください。すると返事を書こうとしているところに指定した引用記号を付けて元の記事の内容が現れるでしょう。但しそのままでは元の記事のヘッダ少しと本文など多くの部分が引用されますので、かなり不要な部分も含まれている可能性があることに注意してください。

あとはメールのときにそうしたように、引用のうちから不要な部分を削除し、残した必要な部分に対するコメントを書き足して行きます。

そうやって記事の本文が書き終わったら、C-c C-c で記事が発信されます。反対に記事を書いている途中で、その記事を投稿する気がなくなった場合は、C-c C-k とすれば、その記事を破棄することができます。

#### 5.4.10 記事のキャンセル

自分で投稿した記事については、その取り消しが可能です。この作業を記事のキャンセルと呼んでいます。例えば投稿した後で、書いた記事の内容に誤りを発見した場合や、投稿するグループを大きく外してしまった時などにキャンセルしたくなると思います。しかしキャンセルは非常手段で、多くのインターネット上の人々に迷惑をかけてしまいます。何よりもまず不注意な記事の投稿をしないように心がけることが大切だということを忘れないでおいってください。

記事をキャンセルするには、図 5.7 の状態のように、キャンセルしたい記事を読んでいる状態でc キーを押します。すると以下のようなメッセージをエコーラインに表示して本当にキャンセルして良いかどうか確認してきます。

```
Do you really want to cancel this article? (yes or no)
```

ここで yes<Return>と答えることによって記事がキャンセルされます。

#### 5.4.11 メールで返事をする

ネットニュースの記事に対する返答というのは、同じく記事によって行なうフォローもあるが、時には記事の投稿者にメールによって返答したくなる場合もあるでしょう。

記事に対してメールで返事をするには、図 5.7 の状態のように、元の記事を読んでいる状態でr キーまたはR キーを押します。r とRの違いは元記事の内容の引用にあります。R は元記事の内容を引用してくれます。引用の際にはフォローの時と同じように引用記号を何にするか聞いてきます。いずれにしても以下のような状態になり、ここでメールを書くわけです。



図 5.9 メールで返事を書く

いつものように--text follows this line--の行から下にメッセージを書きます。この行を消してはいけません。内容の編集については普通の Mule のつもりで操作が可能です。メールの中身を書き終わったら C-c C-c でメールを送信します。反対にメールを書いている途中で、そのメールを送信する気がなくなった場合は、C-x k とします。するとエコーラインに以下のような確認の為の質問をしますので、<Return>とします。

Kill buffer: (default \*mail\*)

これで、そのメールを破棄することができます。

#### 5.4.12 古い記事を読み返す

GNUS は、未読記事が一つもないとそのニュースグループをグループモードでは表示しなくなり、記事モードでも普通では未読のものしか表示しません。一度読んでしまったニュースグループから、古い記事を再び読み返したい時は以下の手順を追うのがもっとも簡単です。

1. (もしあれば) そのニュースグループの記事を全部読んでしまう。未読の記事をゼロにする。
2. (もしグループリストに表示されていないが、ニュースグループの名前を覚えていれば) j キーでニュースグループ名を指定して表示させる。
3. (もしグループリストに表示されていなければ) L キーでそのニュースグループを表示させる。
4. グループモードで未読ゼロのニュースグループを選択し、読む。

要するに GNUS は未読記事ゼロのニュースグループを読むという指示をしてやると、以前の記事を読み返せるように働いてくれるのです。残っている記事が非常に多いと、その中から最近のものをいくつか選んで表示するかを以下のようなメッセージで問い合わせてきます。

How many articles from sandai.general (default 197):

例では sandai.general ニュースグループの記事が結構多いので、そのうちいくら取り出すかを聞いています。どうやら全部で 197 記事あるようで、黙って <Return>すると 197 取り出してくれます。つまり自分が読み返したいと思う記事の数をタイプしてください。すると指定した数だけの既読記事が取り出され、図 5.7 のように表示されるでしょう。あとは今まで通りの操作で同じように扱えます。

### 5.4.13 記事の保存

ほとんど全てのネットニュースの記事は時間が経てばシステムの中から自動的に削除されてしまいます。つまり残しておきたい記事に関しては自分の手元に保存しておく必要があるのです。

記事を保存するには図 5.7 の状態で記事を読んでいる時に、C-o キーを押します。するとエコーラインに以下のようなメッセージを表示して何という名前で保存するかを聞いてきます。

```
Save article in Unix mail file: (default sandai.test) ~/News/
```

上の例ではたまたま sandai.test ニュースグループの記事を保存しようとしています。あなたのホームディレクトリの下で News というディレクトリの下に sandai.test という名前のファイルとして保存すると言っていますので、ここではそれに従うことにします。単に <Return> です。もしもファイルの名前を変更したければここでファイル名をタイプして <Return> です。ファイルが既存であればどんどん追加書きしていきます。

```
Loading rmailout...done
```

というメッセージが表示されて、記事が保存されます。

保存された記事をうまく読み返す方法もあるのですが、ここでは余り説明しません。保存したファイルは、特に指定をしなければあなたのホームディレクトリの下で News ディレクトリ以下にニュースグループごとの名前を付けて保存されているでしょうから、このファイルを直接見てください。

Mule の操作に慣れている人の為に

記事を保存する時に C-o ではなくて o キーで保存すると、記事はまるでメールのようにして保存されます。こうして保存された記事は MHE で読み返す事もできます。M-x mh-rmail で MHE を起動して、M-f で保存した記事のニュースグループをフォルダとして指定してやれば良いのです。MHE におけるフォルダの概念に付いてはここでは説明しません。

o で保存された記事は ~/Mail/sandai.test/1 などという名前のファイルとして保存されています。

但しこうやって作成されたファイルは漢字コードが cc 環境標準の EUC ではなくて JIS になっている<sup>24</sup> ので、普通に cat コマンドや more コマンドでその中身（特に漢字の部分）を見る事ができません。中身を直接確認するには Mule で読み込んで見てください。メールが MHE によって保存される時も、このようになっています。

### 5.4.14 ちょっとしたテクニック

記事を選ぶ時に表題一覧を画面いっぱいに表示する

図 5.7 の様に、記事を表題を見ながら選んでいく時に表題一覧が数行分しか表示されなくて操作が面倒な時があります。そんな時は「=（等号、イコール）」キーが有効です。

図 5.6 の状態で、ニュースグループを選ぶ時に <Space> キーを押す代わりに = キーを押してやります。するとそのグループの表題一覧だけが画面いっぱいに表示されます。ここで記事を選んで <Space> キーを押せば普通に記事を読むことができます。

図 5.7 の状態でも、= を押してやれば、画面いっぱいに表示されます。

---

<sup>24</sup> この表現が判らない人は無視してください。なにか普通ではないファイルになっているとだけ思ってください。

あるグループの記事を全部読んだことにする

GNUS は記事を全て未読と既読に区別し、普通に記事を読もうとすると未読のものだけを表示します。そうすることによって GNUS を起動する度に、新たに届いた記事にだけ注目して読み続けることができるのです。

このような仕掛けは毎日ネットニュースを読んでいるような場合には有効ですが、しばらくネットニュースを読まなかったり新しいニュースグループを選んで読みはじめる時には困ることもあります。つまり既にそのニュースグループには記事が数百もたまっていて、とてもではないけれど読み切れないというような事が起こるのです。

そのために GNUS には「今ある記事を全部読んだことにする」という機能があります。グループモードでも記事モードでもいづれでも構いませんから c キーを押せば、今選んでいるグループの記事を全部まとめて読んだことにしてくれます。

c キーを押せばエコーラインに以下の様なメッセージを表示して確認してきます。

```
Delete all articles not marked as unread? (y or n)
```

まるで記事が全て削除されてしまうかのような表示ですが、単に「未読マークを消すよ」と言っているだけなので、安心して y キーを押してください。

#### 5.4.15 signature

128 ページ、5.3.12 でも紹介した signature はネットニュースの記事でも使われます。カッコいい、またはかわいい signature を作ればネットニュースで人気者になれるかもしれません。

#### 5.4.16 記事を書くときの注意

ここでは操作方法ではなく、記事の中身の書き方やその周辺の事柄について説明します。

どこのニュースグループに投稿するか？

ニュースグループは大きく分けて、3 つあります。

sandai.\*<sup>25</sup>

京都産業大学内のみのニュースグループ

配布範囲 (以下 Distribution) は local を選択します。

fj.\*,japan.\*,tnn.\*など

主に日本語だが対象は世界中のネットワーク

Distribution は world を選択します。

fj.\*,japan.\*,comp.\*,rec.\*,soc.\*,sci.\*など

世界中のネットワーク

Distribution は world を選択します。

初めのうちは学内のニュースグループで練習をしてから学外のニュースグループを利用するようにしましょう。なお、上記の他にも海外のニュースグループがいくつかあります。

モデレータがいるニュースグループとないグループ

- モデレータがいるニュースグループは moderated と呼ばれています。このニュースグループに投稿すると、記事はそのニュースグループを管理している人に自動的にメールとして送られ、その人が

有益であると考えた記事のみが全体に投稿されます。どのニュースグループが moderated かどうかは、fj に関しては、ときどき fj.news.lists にポストされる fj のニュースグループに関する記事を参照してください。

- モデレータ（そのニュースグループの管理人）がないニュースグループは unmoderated と呼ばれています。すなわち、投稿すれば必ず全体に投稿されます。

### sandai.test, fj.test の使いかた

sandai.test はポストやフォローのテストを行なうためのニュースグループです。fj.test もその為のものなのですが、fj.test を用いると、fj を購読している 1000 を越えるサイトに記事が流れることとなります。fj.test はシステム管理者が使うべきものですから、あなたは fj.test を利用してはいけません。テストには sandai.test を利用してください。

### 記事の内容について

- 特にこのように書けというものはありませんが、自分の書いた記事は他人が読むことを考えて、読む人の立場に立って書きましょう。ときどき fj.questions. で始まるニュースグループなどで、質問の内容がつかめないものがあります。
- ネットニュースのような文字による通信では、普段相手と会って話す時や電話で話す時と異なり、微妙なニュアンスが伝わりません。時々これに注意しなかったばかりにインターネットで喧嘩をしていることがあります。注意しましょう。
- こう言ったことにならない為にも、しばらく人の投稿を見てネットニュース上でのマナーを学ぶのがいいと思います。喧嘩の元になるようなネットニュース上での「ひどい」記事は fj.news.usage というニュースグループなどでよく取り上げられますので参考にしてください。
- Subject の内容は、その記事の内容を反映しているものにすべきです。例えば、sandai.question という質問のためのニュースグループに、「HELP ME」とか「Question」などという Subject は意味を持ちません。しかし「about fishing」などとすれば釣りに関する質問であることがわかります。後から記事を読み返すときにもわかりやすいですし、そのような Subject をつけるように心がけましょう。また、ネットニュースを読んでいるすべての人が漢字を表示できるコンピュータを使っているとは限りませんので、Subject には漢字を使ってはいけません。
- 投稿の内容に関する責任は、書いた人の属する組織にあるのではなく、書いた人自身にあります。また、著作権はその記事を書いた人にあります。
- ネットニュースにおける誰かの発言は、発言者個人のものであって決して組織の代表としての発言ではないことに常に注意しておいてください。例えば某メーカーの社員が自社の製品について批判的なことを書いても、それは個人の感想に過ぎず、それ以上の意味はありません。その発言を取り上げてその会社の他の人に文句を言ったりするのは「反則」です。自分が買った製品が気に入らなくて「×社の人説明してください」などと発言するのもナンセンスです。むしろこのような場合は「×社の製品を買った人、良い解決法を考えましょう」とやる方が余程建設的です。
- 違法な内容の記事（歌詞のポストなどによる著作権の侵害、公道を時速 で走ったなど）をポストしてはいけません。当たり前の話ですが、反社会的な記事を投稿するべきではありません。それが問題として取り上げられ、インターネットが社会から迫害されてしまう可能性だってあるのです。

## フォローの方法

ネットニュースを読んでいるとわかりますが、相手の引用文の始めには「>」等をつけます。また、引用の前には、誰に対する引用なのかをはっきりさせます。引用の量はなるべく少なくなるようにしてください。なお「>」で始まる行の行数は、自分がポストする記事全体の半分以下でなければなりません（ネットニュースリーダのデフォルトではこうなっています）。つまり引用大量、意見少量という事はするべきでない、という訳です。

## キャンセルについて

自分のポストした記事のみキャンセルを行なうことができます。ただし、キャンセルを行なうと、キャンセルを行なうためのコントロールメッセージが fj.\* のニュースグループならば、fj のすべてのサイトに送られます。キャンセルを行なう時は十分慎重に行なってください。また、最初からキャンセルを行なわなくも済むようなポストをすべきです。

## 引用のマナー

自分のところに送られてきたメールをネットニュースに引用する場合には、メールの送り手の許可を得てからにしましょう。また、ネットニュースを引用する場合には、その引用した記事を特定できるような情報を付けてください。

## 5.5 ファイルをコンピュータ間で移動

家のパソコンで入力した文章を UNIX コンピュータで使いたい。そういうこともあると思います。学校の UNIX コンピュータと自宅のパソコンの間でファイルをコピーするにはどうすればいいのでしょうか。もっとも簡単なのはフロッピーディスクを介してコピーすることです。

さて、本題に入る前にいくつか予備知識が必要になるのでそれを先に説明します。知っている人は読み飛ばしてください。

### 5.5.1 コンピュータウイルス

32 ページ、2.6 で述べたようにフロッピーディスクを持ち込むことはコンピュータウイルスの危険性を伴います。十分に注意してください。

### 5.5.2 フロッピーディスクの種類

フロッピーディスクは外見が同じでも実はいろいろな種類があります。

買ってきたフロッピーディスクはそのままでは使うことができず、初期化<sup>26</sup>が必要になります<sup>27</sup>。初期化とはフロッピーディスクをまっさらな状態にして<sup>28</sup>、コンピュータがデータを読み書きする時のための目印をつける作業です。その印の付け方、これをフォーマット形式といいますが、これがコンピュータの種類によって違うため、あるコンピュータで使えたフロッピーディスクが他のコンピュータで使えないということが起こります。ただ最近のコンピュータはいくつかのフォーマット形式を扱えるようになっていて学校のコンピュータと家のコンピュータ両方で扱えるフォーマット形式があるはずですが、ではどんなフォーマット形式があるか紹介しましょう。

表記	説明
PC-AT	IBM PC-AT 互換機。俗に DOS/V パソコンと誤って呼ばれているもの。10 号館情報処理教室の Linux や WindowsNT, 51, 52 情報処理教室にあるコンピュータはこの種類。
PC-98	NEC PC-9801 シリーズとその互換機。但し PC-98NX シリーズは PC-AT 互換機に属する。
Mac	Apple Macintosh シリーズとその互換機。10206, 11, 31, 32 情報処理教室にあるコンピュータはこの種類。
Ultra	21 情報処理教室の Sun Ultra1。

表 5.1 コンピュータの表記

メディア	説明
3.5" (インチ) 2HD	両面高密度倍。なんのことも判らなくても大丈夫。たくさん記録できて、現在最も普及しているフロッピーディスク。
3.5" 2DD	両面倍密度倍。2HD の一世代前のフロッピーディスク。2HD の半分しか記録できないと思えばいいでしょう。

表 5.2 フロッピーディスクの種類

<sup>26</sup> コンピュータ用語でフォーマットともいいます。

<sup>27</sup> 特定の機械用に初期化されたフロッピーディスクも売っています。それならわざわざもう一度初期化する必要はありません。また、間違っても他の機械用に初期化されたフロッピーディスクを買ってしまったら、目的のコンピュータで初期化してやることで普通は使えるようになります。

<sup>28</sup> 大事なファイルが入っていた場合でも全部消してしまいますので実行するときはよく確認しましょう。

種類	容量	通称	利用できる主なコンピュータ
2HD	1.44MB	1.44 メガニイエチディー, ドスブイフォーマット	PC-AT, 一部の PC-98, Mac, Ultra
2HD	1.44MB	マッキントッシュユニイエチディー	Mac
2HD	1.44MB	UNIX ニイエチディー	Ultra
2HD	1.25MB	1.2 メガニイエチディー, 98 フォーマット	一部の PC-AT, PC-98
2DD	800KB	マッキントッシュニディーディー	Mac
2DD	720KB	720 ニディーディー	PC-AT, PC-98, Mac, Ultra
2DD	640KB	640 ニディーディー	PC-98

表 5.3 主なフォーマット形式一覧

どうですか？たくさんありますね。しかしあなたが使うことになるのはどれか一つです。

- 自宅のパソコンが PC-AT 互換機または 1.44MB 2HD が使える PC-98 シリーズの場合  
1.44MB 2HD を使います。自宅から学校へ持っていく場合には何の問題もありません。逆の学校から家へは学校のコンピュータが UNIX や Macintosh の場合、ファイル名の長さ制限に気をつけましょう。扱えるのは 8 文字.3 文字 までです。
- 自宅のパソコンが 1.44MB 2HD の使えない PC-98 シリーズの場合  
少し古いタイプの PC-98 シリーズですね<sup>29</sup>。この場合 720KB 2DD を使った方がよさそうです。制限は上の 1.44MB 2HD の場合と同じです。
- 自宅のパソコンが Macintosh シリーズの場合  
1.44MB Macintosh 2HD を使えばよいでしょう。学校の Macintosh を使ってデータをやりとりする場合は何の問題もありません。学校側が Windows パソコンなら 1.44MB 2HD を使いましょう。同じくファイル名の長さ制限に気をつけましょう。

このようなところでしょうか。それぞれの初期化の仕方はパソコンやワープロのマニュアルを参照してください。

### 5.5.3 ファイルの種類

#### テキスト形式とバイナリ形式

さてフォーマット形式は理解できたでしょうか？できたらついでにもう一つ覚えましょう。ファイルの種類です。ファイルにはテキスト形式とバイナリ形式の 2 種類があります。

- テキスト形式  
Mule で編集したファイルなど、文字だけのものです。
- バイナリ形式  
プログラムデータ等、文字以外のものを含むデータです。mule で開けようとしたり、cat で画面に表示しようとするときとひどい目にあいます<sup>30</sup>。

勘違いしやすいのですが、パソコンのワープロソフト（一太郎やワード等）で保存したファイルはバイナリ形式です。いろいろな修飾データ（文字を大きくしたり下線を引いたりする情報）を保存するためです。パソコンのワープロソフトでテキスト形式のファイルを作成したい場合は保存するときにテキスト形式で保存するようにしてください<sup>31</sup>

<sup>29</sup> 筆者の家にもありますが (^\_^;)

<sup>30</sup> 画面が乱れてそれ以降正しい表示ができなくなる場合があります

<sup>31</sup> 詳しくはソフトのマニュアルを見てください。また、この場合拡大文字や下線などの情報が保存できません。

## テキスト形式の場合

テキスト形式は行を変えるときに、目には見えない改行コード（改行を意味する文字）が存在しています。この改行コードはコンピュータの種類毎に違いがありこの扱いが違うコンピュータ間でファイルをやり取りするときはこれを変換する必要があります。改行コードの違いをまとめると表 5.4 のようになります。

改行コードの種類	主な OS
0A	UNIX
0D	Macintosh OS
0D 0A	Windows95, Windows3.1, MS-DOS

表 5.4 改行コードの種類

表中の改行コードの意味は深く考えないでいいです。簡単に説明すると、たとえば UNIX で作ったテキストデータを Windows95 にそのまま持っていったとしましょう。UNIX で作った文書には改行するのに 0A としか書かれていません。しかし Windows95 で改行するには 0D と 0A が並んで存在する必要があります。従ってこれは改行コードとはみなされず、正しく改行がなされないことになってしまうのです。また逆の場合には 0D が余分なため変な文字が表示される場合があります。

サンプルデータ	貴社の記者が 0A 車で帰社した	
各 OS での扱い	UNIX の場合	Windows95 の場合
	貴社の記者が 車で帰社した	貴社の記者が車で帰社した
サンプルデータ	貴社の記者が 0D 0A 車で帰社した	
各 OS での扱い	UNIX の場合	Windows95 の場合
	貴社の記者が^M 車で帰社した	貴社の記者が 車で帰社した

改行コードの他にもう一つ、文字コードの違いもあります。文字コードとは日本語を扱う時のコンピュータの情報の持ち方を表したものです。コンピュータはアルファベットや記号やひらがなや漢字等、全ての文字をずらっと並べて 1 番目の文字、2 番目の文字、というように覚えているのです。

その並べかたが何通りかあり、それを文字コードと呼びます。例えば「あいうえお」という文字が、コンピュータ A の文字コードでは 1000 番目、1001 番目、...、1004 番目の文字として、コンピュータ B の文字コードでは 2000 番目、...、2004 番目の文字として記録しているとします。この場合、コンピュータ A で「あお」と書いて記録したテキストをコンピュータ B で呼び出しても、コンピュータ B はあくまでコンピュータ B の文字コードでの 1000 番目と 1004 番目の文字を表示しようとするだけで、「あお」とは表示できません。仮にコンピュータ B が 1000 番目からの文字は A B C D E となっていたら「A E」と表示されてしまいます。

コード番号	コンピュータ A	コンピュータ B	
:	:	:	
1000	あ	A	「あいうえお」という文書を
1001	い	B	・コンピュータ A で作った場合
1002	う	C	1000 1001 1002 1003 1004
1003	え	D	と記録される。
1004	お	E	・コンピュータ B で作った場合
:	:	:	2000 2001 2002 2003 2004
2000	亜	あ	と記録される。
2001	唾	い	・コンピュータ A で作った
2002	娃	う	「あいうえお」を
2003	阿	え	コンピュータ B で表示すると
2004	哀	お	「A B C D E」に見える。
:	:	:	

このようにコンピュータが扱う文字コードとファイルに使われている文字コードが違った場合は全く使い物にならないのです。でも安心してください。文字コードを変換してくれるコマンドが存在しますので、それを利用して目的の文字コードに直すことができます。

文字コードには表 5.5 のように 3 つの代表的なものがあります。覚えておいてください。また、本ガイド中ではあえて JIS コード、EUC コード、SJIS コードと表記することにします。

文字コードの名称	簡単な説明
ISO-2022-jp	一般的に誤って JIS コードと呼ばれるもの。主に電子メールなど、ネットワークで用いられる。
EUC-JP	一般的に EUC コードと呼ばれるもの。主に UNIX ワークステーションで用いられる。
Shift JIS	SJIS コード、MS 漢字コードと呼ばれることもある。主に PC-AT や PC-98、Macintosh 等のパソコンで用いられる。上記 JIS X 0208 の各規格から変換規則に従って定義されているため JIS X 0208 と同じ数の種類がある。

表 5.5 代表的な文字コード

cc 環境では EUC コード、一般的なパソコンが SJIS コードを使っているため、テキスト文字をコピーしてもそのままでは読むことができないはずです。

これを変換してやらないといけません。cc 環境では `nkf`<sup>32</sup> というコマンドが用意されています。利用方法は

```
nkf -変換したい文字コード 元のテキストファイル名 > 変換後のテキストファイル名
```

です。文字コードには `s`(SJIS)、`j`(JIS)、`e`(EUC) が選べます。例えば `cc2000` で `report2` というファイルを作り (EUC)、これを家のパソコンで読むために SJIS コードに変換するには

```
cc2000(80)% nkf -s report2 > report2.txt
```

<sup>32</sup> Network Kanji code conversion Filter から命名されているらしいです。

のようにします。もしくはそのまま持ち帰り、パソコン上で同様のソフトを使って変換すればよいでしょう。注意点として、UNIX では変換後のファイル名に元のファイル名と同じ物 (例では report2) を指定するとファイルがからっぽになってしまいます。これは UNIX の処理方法を理解するとわかるのですが、UNIX はまず書き込むファイル report2.txt を用意して、その後 report2 を変換しながら report2.txt に書き込んでいきます。従って出力先を同じファイル名、例えば `nkf -s report2 > report2` とすると、書き込むファイルを用意する時 report2 をからっぽにします。それはすなわち元文書である report2 を消してしまうということです。結果、中身のなくなった report2 を変換して書き出すので、当然中身は何も残りません。

#### 5.5.4 ファイルをフロッピーディスクにコピー

パソコンのファイルをフロッピーディスクにコピーすることはできると思います<sup>33</sup>。ここでは各情報処理教室で UNIX コンピュータのファイルをフロッピーディスクに読み書きする方法を紹介します。また、いくつかの情報処理教室では同様に MO ディスクを扱うことができます。フロッピーディスクと同様にお使いください。

- 10206、11、32 情報処理教室の MacOS で Fetch を利用する
- 21 情報処理教室で cp コマンドを利用する
- 10 号館情報処理教室の Linux で cp コマンドを利用する
- 10 号館情報処理教室、または 51 情報処理教室の WindowsNT で WS FTP を利用する

#### 10206、11、32 情報処理教室の MacOS で Fetch を利用する

11、31、32 情報処理教室の Macintosh では Fetch というソフトを使います。

アップルメニューから fetch という項目を選びます。🖱️のようなアイコンが描いてあると思います。これを選ぶと Host、USER ID、Password を入力する画面 (図 5.10 左側) が出てくるので、それぞれ cc2000、あなたのユーザ ID、あなたのパスワードを入力して OK ボタンを押します。パスワードは何を入力しても「\*」が表示されると思いますが、これは後ろから覗かれても大丈夫なようにしているだけです。安心して入力してください。きちんと入力できていれば図 5.10 右側のような画面になります。

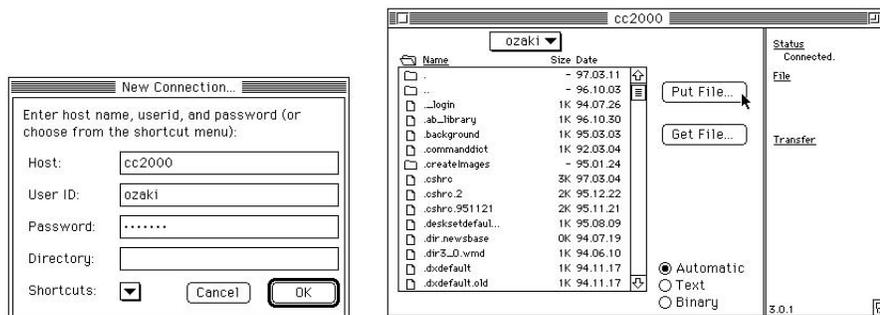


図 5.10 Fetch の起動

<sup>33</sup> え、わからない? パソコンのマニュアルを見てください。

・ Macintosh のフロッピーディスクから UNIX コンピュータへ

PUT ボタンを押します。するとコピーしたいファイルを選ぶ画面になりますので選択してください。選びたいファイルが見当たらない場合はフロッピーディスクではなく、ハードディスクが表示されているのかもしれませんが。その場合、まずフロッピーディスクを選択しなければいけませんので、デスクトップというボタンを押すと一覧の中にフロッピーディスクが出てきますからそれをダブルクリックするといいでしょう。



無事目的のファイルを選ぶことができたなら、図 5.11 のようになります。ここで Text タイプとなっていますが、これは必要に応じて Raw Data に変更する必要があります。テキスト形式のデータなら Text を、バイナリ形式のデータなら Raw Data を選択してください。

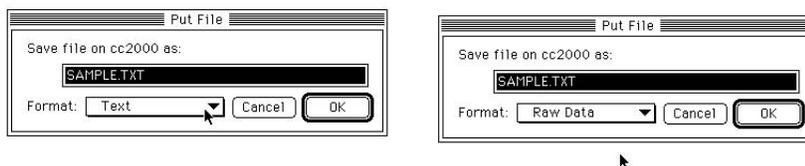
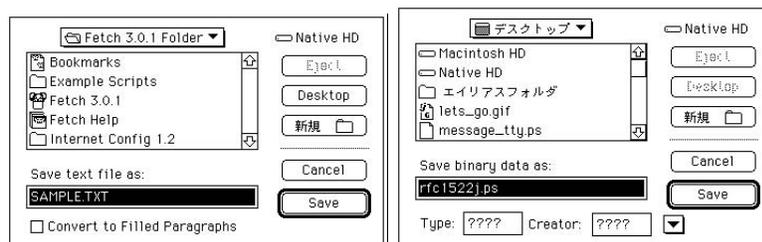


図 5.11 データ形式の選択

・ UNIX コンピュータから Macintosh のフロッピーディスクへ

同様に Fetch を起動して、今度は先にファイル形式を決めます。ファイル形式にあわせて、Automactic になっているボタンを Text か Binary にあわせてください。続いて表示されているファイルリストから必要なファイルを選択し、Get ボタンを押します。すると保存場所を聞いてきます。Text を選んだ時と Binary を選んだ時では若干画面が違いますが、気にせずフロッピーディスクを選択して保存してください。



## 21 情報処理教室で cp コマンドを利用する

21 情報処理教室の Ultra1 では UNIX から 1.44MB(DOS/V フォーマット) のフロッピーディスクに cp コマンドで読み書きすることができます。

フロッピーディスクを挿入して、ルートメニューから、Floppy Mount を選択すると /floppy というディレクトリにフロッピーディスクが接続されます。このディレクトリに対して例えば cp report2 /floppy/report2 のように cp, rm, mv 等のコマンドが使用できます。

取り出す時は同じくルートメニューから、Floppy Eject を選択してください。なお、この時にカレントディレクトリが /floppy 以下にあると「使用中です」とメッセージがでます。その場合、カレントディレクトリを移動して再度 Floppy Eject してください。よく判らなければ一度セッションを終了して login しなおすといいでしょう。

21 情報処理教室のいくつかのコンピュータでは Transfer Pro というソフトウェアが使えます。これを使うと、MO ディスクとフロッピーディスクに対して比較的簡単に UNIX のファイルをコピーすることができます。詳しくは 21 情報処理教室にある手引きを参照してください。

## 10 号館情報処理教室の Linux で cp コマンドを利用する

Linux から 1.44MB(DOS/V フォーマット) のフロッピーディスクに cp コマンドで読み書きすることができます。

フロッピーディスクを挿入して、mount /mnt/floppy とコマンドを実行してください。すると/mnt/floppy というディレクトリでフロッピーディスクにアクセスできます。普通に cp report2 /mnt/floppy/report2 のように cp, rm, mv 等のコマンドが使用できます。

利用後、そのままフロッピーディスクを取り出すとフロッピーディスクの内容を壊してしまう可能性があるため、取り出す前には umount /mnt/floppy とコマンドを実行してからフロッピーディスクドライブのイジェクトボタンを押して取り出してください。

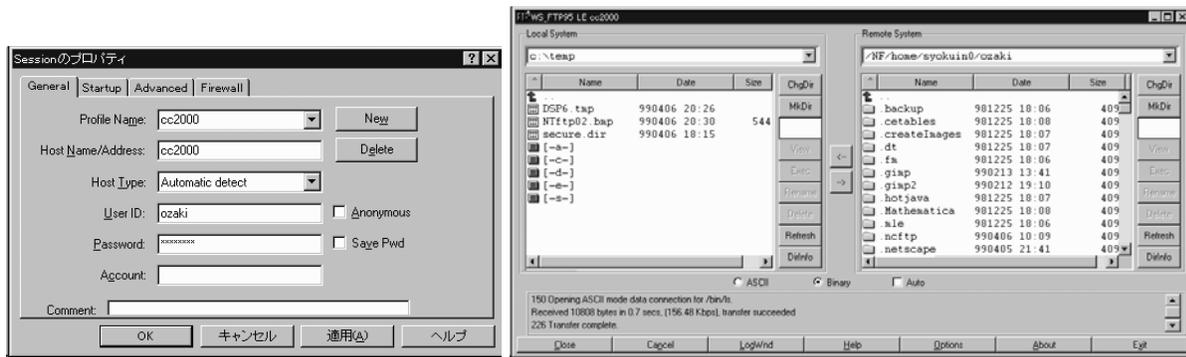
なお、umount コマンド実行時にカレントディレクトリが /mnt/floppy 以下にあると失敗します。その場合、カレントディレクトリを移動して再度 umount してください。よく判らなければ一度セッションを終了して login しなおすといいでしょう。

## 10 号館情報処理教室、または 51 情報処理教室の WindowsNT で WS\_FTP を利用する

WindowsNT では WS\_FTP を使います。



共通アプリケーションからプログラムの WS\_FTP95 LE を選択するとユーザ ID、パスワードを求める画面が出てきます。左側の図のように、HostName に cc2000、UserID に cc 環境の UNIX のユーザ ID、Password に cc 環境の UNIX のパスワードを入力し、OK ボタンを押すと右側の図のように、左側にパソコンのファイル、右側に UNIX のファイルが表示された画面が出てきます。



ディレクトリ名はダブルクリックするとそのディレクトリに移動します。フロッピーディスクを選択するにはパソコン側のディレクトリ名一覧に [-a-] として A ドライブ、つまりフロッピーディスクがありますのでこれをダブルクリックします。

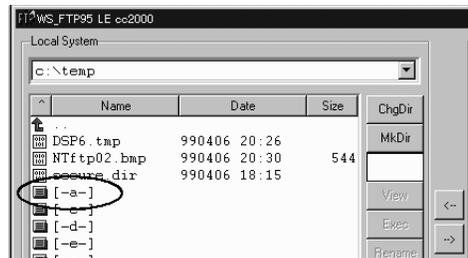


図 5.12 フロッピーディスクの選択

パソコン側、UNIX 側それぞれの目的のディレクトリに移動して、下段のファイル名一覧から目的のファイルをクリックして色を反転させます。そしてファイルにあわせて ASCII か Binary のどちらかに設定し、「パソコンから UNIX」なら「->」を、「UNIX からパソコン」なら「<-」のボタンを押すと転送を始めます。

### 5.5.5 UNIX コンピュータ間でファイルのコピー

もし、あなたが 2 種類以上の UNIX コンピュータ環境を利用していた場合、それらのコンピュータ間でファイルをコピーするのならばフロッピーディスクは必要ありません。

FTP(File Transfer Protocol)、日本語でいうならファイルを転送するためのルール、とでもなるでしょうか。違う UNIX コンピュータでもこの FTP に従って通信することでファイルのコピーができるのです<sup>34</sup>。UNIX コンピュータには ftp コマンドという、FTP を使ってファイルをコピーするコマンドが用意されています。

例えば ozaki さんは cc2000 と ksunny という 2 つのコンピュータを利用しています。ここで ozaki さんが cc2000 を使っていて、ksunny というコンピュータの reidai.txt というファイルが欲しくなったとしましょう。まず cc2000 側で

```
cc2000(80)% ftp ksunny
```

<sup>34</sup> 実は先の章の Fetch や WS\_FTP もこれを利用しているソフトです。

とすると

```
Connected to ksunny.  
220 ksunny FTP server (UNIX(r) System V Release 4.0) ready.  
Name (ksunny:ozaki):
```

となり、ユーザ ID とパスワードを求めてきますので ksunny でのユーザ ID とパスワードを入力します<sup>35</sup>。  
正しく入力できたら

```
User ozaki logged in.  
ftp>
```

となり、命令の入力待ちの状態になります。

どのようなコマンドが使えるかは help <Return>とすると一覧が出ます。とりあえずディレクトリを移動する cd、ファイルの一覧を見る ls、コピー時にテキスト形式で転送するか、バイナリ形式で転送するかを選択する ascii と bin、ファイルを取得する get、ファイルを置く put、そして終了する quit。これだけ覚えていれば使えます。

最初は ozaki さんの ksunny でのホームディレクトリにいるはずですから、cd コマンドで目的のファイルのあるディレクトリに移動します。必要なら dir コマンドで目的のファイルを確認し、ファイルの種類にあわせてテキストなら ascii、バイナリなら bin コマンドを実行して、get reidai.txt とすれば cc2000 で ftp コマンドを実行したときのカレントディレクトリに reidai.txt をコピーします。終了は quit コマンドです。

```
ftp> cd dir1  
250 CWD command successful.  
ftp> ascii  
200 Type set to A.  
ftp> get reidai.txt  
200 PORT command successful.  
150 ASCII data connection for reidai.txt (133.101.32.73,48069) (635 bytes).  
226 ASCII Transfer complete.  
local: reidai.txt remote: reidai.txt  
648 bytes received in 0.004 seconds (1.6e+02 Kbytes/s)  
ftp> quit  
221 Goodbye.  
cc2000(81)%
```

## FTP コマンドの詳細

ここでよく使うものをまとめておきましょう。

---

<sup>35</sup> ここで他人のユーザ ID、パスワードを入力するとその人のファイルが FTP の対象になります。友人のファイルをコピーする時に使うといいでしょう。

ディレクトリ関連	cd pwd ls dir lcd	ディレクトリを移動します。 現在のディレクトリを表示します。 現在のディレクトリのファイルの情報を表示します。 現在のディレクトリのファイルの情報を詳しく表示します。 自分の使用しているコンピュータ側のディレクトリを移動します。
ファイルの種類	bin ascii	ファイルの転送モードをバイナリ形式にします。 ファイルの転送モードをテキスト形式にします。
ファイルの転送	get filename put filename mget file1 file2 ... mput file1 file2 ...	ファイルを受信します。 ファイルを送信します。 複数のファイルを受信します。 複数のファイルを送信します。
ヘルプ	prompt off help	mget,mput の際に確認を取りません。 コマンドの一覧を表示します。
終了	quit	ftp を終了します。

### 5.5.6 anonymous FTP

今までの説明は自分のファイルをコピーする方法でした。実はこの応用で、インターネット上に提供されている様々なファイルを取得することもできるのです。

インターネット上のファイルを取得する場合はユーザIDに anonymous(匿名)を使うのでこれを anonymous FTP と呼びます。ユーザIDに anonymous、パスワードにメールアドレス<sup>36</sup>を入力して利用します。パスワードにメールアドレスを入力するのが anonymous FTP を使わせてもらう上でのマナーです。違う文字列を入力してもおそらく受け付けますが、ここはマナーを守ってきちんとメールアドレスを入力してください。きちんと入力しておくことでFTPによって得たものにウイルスなどの疑いがあれば連絡してくれるでしょう。パスワードの入力部分ですので入力した文字列は画面に表示されません。慎重に入力してください。

また、ファイルを転送している間はその相手先と自分との間のネットワークをいくらか占有することになります。ネットワークはみんなで共有しているものです。したがって大きなサイズのファイル<sup>37</sup>を転送すること(長時間占有することになります。)、遠くからファイルを転送すること(長距離の回線を占有することになります。)はお勧めできません。どうしても大きなファイルを得たいときは夜などのすいている時間帯に実行するか、同じファイルをいくつものサイト<sup>38</sup>にコピーして置いてある場合<sup>39</sup>がありますので、できるだけ近くのFTPサイトから得るようにするのがマナーです。ちなみに本学の場合なら一番近いのは本学内の ftp.kyoto-su.ac.jp 等、次は京都大学の ftp.kuis.kyoto-u.ac.jp です。

本学の ftp サービスのディレクトリ構成は以下になっています。

/pub

/comm	UNIX 用シリアル通信関係
/net	UNIX 用 LAN サービス関係
/util	UNIX 用汎用のユーティリティ
/fep	かな漢字変換

<sup>36</sup> パスワードにあなたの本当のパスワードを入力してはいけません。もしも本当のパスワードを入力してしまったらすぐにパスワードを変更してください。

<sup>37</sup> 色々な条件によって違いますが、500KB ぐらいを目安にすればいいと思います。

<sup>38</sup> サイトとは直訳すれば敷地ですがここではネットワークのまとまった単位です。例：京都産業大学のサイト

<sup>39</sup> ミラーサイトと呼ばれます。

/doc	いろいろなドキュメント
/X	X ウィンドウ関係
/GNU	GNU 関係
/TeX	TeX 関係
/DOS	DOS 関係
/Windows	MS-Windows 関係
/OS2	OS2 関係
/Mac	Mac 関係
/NeXT	NeXT 関係
/PC98	PC98 関係 (DOS 汎用は ./DOS 行き。)
/Solaris2	Sun Solaris2 関係
/AIX	IBM AIX 関係

各アーキテクチャ(Windows, Mac 等)以下の構成(主なもの)

/AntiVirus	ウイルス駆除ソフト関係
/DiskTools	圧縮解凍ソフト、ファイル操作関係
/Editor	エディタ関係
/Graphics	グラフィック関係
/NetTools	ネットワーク関係
/Others	その他

例えば本学の FTP サイトを利用して、/pub/doc/report\_j.tex というファイルを取得する場合はこのようになります。

```
cc2000(80)% ftp ftp.kyoto-su.ac.jp
Connected to cc2000.kyoto-su.ac.jp.
220 cc2000 FTP server (Version wu-2.4(1) 1994 年 11 月 15 日 (火) 09 時 46 分 10 秒
JST) ready.
Name (ftp.kyoto-su.ac.jp:ozaki): anonymous
331 Guest login ok, send your complete e-mail address as password.
Password: ozaki@cc.kyoto-su.ac.jp40
230-#####
230-# Welcome to the Kyoto Sangyo University Computer Center archive.
230-# All archive files are stored under /pub.
230-# Please report if you have problem to ozaki@cc.kyoto-su.ac.jp
230-#####
230-
230 Guest login ok, access restrictions apply.
ftp>
```

となり、命令の入力待ちの状態になります。最初は FTP サービスのルートディレクトリにいますから、cd コマンドで目的のファイルのあるディレクトリに移動します。そして今回の例では取得するファイル形式がテキストなので ascii と宣言して、get report\_j.tex とすれば cc2000 で ftp コマンドを実行したディレクトリに report\_j.tex をコピーします。最後は quit コマンドで終了します。

<sup>40</sup> 実際はここは画面には表示されません。

```

ftp> cd /pub/doc
250 CWD command successful.
ftp> ascii
200 Type set to A.
ftp> get report_j.tex
200 PORT command successful.
150 Opening ASCII mode data connection for report_j.tex (6086 bytes).
226 Transfer complete.
local: report_j.tex remote: report_j.tex
6294 bytes received in 0.041 seconds (1.5e+02 Kbytes/s)
ftp> quit
221 Goodbye.
cc2000(81)%

```

### 5.5.7 archie

ファイルの名前は分かるんだけど、どこの FTP サイトにあるか分からないという時にはそれを検索する、archie<sup>41</sup> というサービスがあります。

```
cc2000(80)% archie -s 'strings'
```

のようにすると（しばらく時間がかかりますが）strings を含むファイル名を検索してそれがどこにあるか表示してくれます。

ただし、先述した通り本学にあるものをわざわざ他の FTP サーバに取りに行くのはネットワーク資源の無駄使いになります。また、この archie では本学の分は検知できませんので、これを行う前に本学の FTP サーバに目的のファイルがないか確認してください。そのために ccarchie<sup>42</sup> というコマンドが用意してあります。使い方は以下のとおりです。

```

cc2000(80)% ccarchie -s 'strings*' strings から始まるものを検索。
cc2000(80)% ccarchie -s '*strings*' strings を含むものを検索。
cc2000(80)% ccarchie -s '*strings' strings で終るものを検索。
cc2000(80)% man ccarchie ccarchie の詳しい使い方を表示。
一例を載せておきます。

```

```
cc2000(173)% ccarchie -s 'unixguide*'
```

```

Location: /pub/doc
  DIRECTORY drwxrwxr-x      4096 Mar 25 1994 UNIXguide
  DIRECTORY drwxrwxr-x      7168 Oct 17 10:22 UNIXguide_2nd
  DIRECTORY drwxrwxr-x      3072 Jul  4 1995 UNIXguide_append
Location: /pub/doc/UNIXguide_2nd
  FILE -rw-rw-r--          915 Apr  7 1995 unixguide.sty

```

これは学内 ftp サーバで提供されているもののうち、unixguide という文字列を含むディレクトリやファイル名が表示されています。例えば、下の 2 行は unixguide.sty というファイルが ftp.kyoto-su.ac.jp に anonymous FTP して、/pub/doc/UNIXguide\_2nd のディレクトリから取得できることを示しています。

<sup>41</sup> アーチーと読みます。

<sup>42</sup> cc 環境で用意されたコマンドです。一般的な UNIX にはありません。

## 5.6 WWW

WWW のことは 113 ページ、5.1.9 でも少し説明しましたが、ここでは使い方も含めてもう少し詳しい説明をしていきます。

WWW という呼び方は正式には World Wide Web<sup>43</sup> と言います。

WWW はブラウザと呼ばれる種類のソフトウェアを使って WWW ページを簡単な操作で見ることができます。WWW ページやブラウザの詳しい説明はもう少し後ですが、簡単に言うと WWW ページは情報でブラウザはその情報を読むための道具と言えます。

ここで WWW の利用方法の一例を紹介します。最初にまず、京都産業大学の WWW ページを見てみましょう。情報処理教室にあるコンピュータのブラウザを使うと京都産業大学の代表となる WWW ページが表示されます。その項目の中で「情報サービス」と書かれている部分をクリックします。そうするとブラウザの表示画面が、情報サービスという題名の画面に切り替わります。情報サービスという題名の画面には「WWW 個人ページを作ろう」や「ネットサーフィン入門」、「産大 FAQ 初級者編」等の項目があり、そこをクリックすると画面がクリックした情報に切り替わります。

このように WWW を利用すると、ひとつの情報から関連するいろいろな情報に移っていくことができます。またそれぞれの WWW ページには、情報の所在を表わす URL (後述) というものがあり、それを直接指定することで別の画面からたどっていかなくても直接その画面を表示することもできます。

WWW を使っているいろいろな場所に行き、さまざまな情報をブラウザに表示することをネットサーフィンと呼びます。この後の章を読むことで自由にインターネット上をネットサーフィンし、豊かなネットワークライフを送れるようになることを目指しましょう。

### 5.6.1 ホームページ

WWW ページは企業や団体、個人が自己PRやサービスの為に作成するもので、利用者は WWW ページからさまざまな情報を入手できます。情報は一つの WWW ページで構成されることもありますが、あまり一つのページに情報を詰め込んでも見にくくなりますから、普通は整理され、複数の WWW ページに分かれて構成されています。そのように複数に WWW ページを分けると、それらを代表する玄関ページができあがります。これをホームページと呼びます。ホームページの規模は様々で、大きなものだと京都産業大学のホームページ (京都産業大学を代表する WWW ページ)、小さなものだと尾崎さんのホームページ (個人が作成しているページの玄関ページ) などが挙げられます。雑誌やテレビ等で誤って全ての WWW ページをホームページと呼ぶ報道がなされたためにそのような意味でホームページと呼ぶ人が多いようです。基本的には代表するページをホームページと呼ぶということを覚えておきましょう。WWW ページが扱える情報は、以前は文字と絵、写真だけだったのですが、今ではそれらに加えて動画、音声なども扱えるようになっていきます。

WWW ページの特徴のひとつに別のページに移動する機能があります。この機能のことをリンクと呼びます。先ほど京都産業大学のホームページから「情報サービス」、そしてそこからいろいろなページに移動する例を挙げました。このように別のページに移動できるのもリンクを使っているからです。先ほどの例ではページをマウスでクリックすると、別のページに移動しましたがどこをクリックしても移動するわけではありません。移動するのはリンク定義がされている場所だけです。リンク定義がされた場所は、文字の色が他と違っていたり、アンダーラインが付いているので慣れれば見分けがつくでしょう。また、文字だけでなく画像上にリンクの定義がされている場合もあります。画像上にリンクが定義されている場合も、マウスを近づけるとカーソルの形が変わり、ブラウザの左下に移動先の URL が表示されますので、や

<sup>43</sup> 直訳すれば「世界規模の蜘蛛の巣」とでもなるのでしょうか。蜘蛛の巣のように張り巡らされたインターネットを利用した情報網だからこのような命名だったのでしょうか。興味があったら調べてみてください。

はり慣れればすぐに分かるようになるでしょう。

## 5.6.2 URL

URL(Uniform Resource Locator)<sup>44</sup> は情報のインターネット上の所在を表わすものです。例えば、産大の日本語のホームページの URL は

`http://www.kyoto-su.ac.jp/index-j.html`

のように表記されます。このときの「`http://`」は WWW のページだということを、「`www.kyoto-su.ac.jp`」はページを持っているコンピュータの名前、「`/index-j.html`」はそのコンピュータ上でのページの所在を表しています。

この URL さえ知っていれば、インターネット上で欲しい情報をすぐに得ることができます。最近雑誌や新聞でも URL の紹介をしているようですし、ホームページを持っている企業はカタログや企業 PR の場で URL を紹介するのが一般的になりつつあります。

## 5.6.3 URL の検索

URL がわからない場合はどうすればいいのでしょうか。電話番号なら 104 で聞いたり電話帳で調べることができますね。WWW の世界にも同様のサービスを行ってくれるものがあります。

- キーワード検索  
104 の電話番号案内に相当するサービスです。キーワードを使って検索を行います。別名サーチエンジンと呼ばれています。
- ディレクトリ検索  
電話帳に相当するサービスです。知りたい情報が階層的にまとめられています。

キーワード検索はキーワードを指定して検索すると、そのキーワードが含まれている WWW ページの一覧を表示してくれるものです。例えば `university` と `kyoto` をキーワードにして検索すると京都産業大学や京都大学などを見つけることができます。

ディレクトリ検索はジャンル選択を繰り返し行うことで検索します。京都産業大学を探す場合は、ジャンルから教育を選び、次に大学を選び、都道府県から京都を選んで京都産業大学を見つけるという感じになります。

キーワード検索で有名なのは `goo` や `Altavista`、ディレクトリ検索で有名なのは `yahoo` や `NTTDIRECT` 等があります。

名前	URL
<code>goo</code>	<code>http://www.goo.ne.jp/</code>
<code>Altavista</code>	<code>http://altavista.digital.com/</code>
<code>yahoo</code>	<code>http://www.yahoo.co.jp/</code>
<code>NTTDIRECT</code>	<code>http://navi.ntt.co.jp/</code>

<sup>44</sup> 最新規格で URI(Uniform Resource Indicator) という名前に変更されました。

## 5.6.4 ブラウザ

WWW を見るためのソフトウェアを WWW のブラウザと言います。Microsoft Internet Explorer や Netscape Navigator などがその代表です。

### 各コンピュータからの利用方法

WWW を利用できる本学の共同施設は現在以下のようになってます。

MacOS	10206,11,32	情報処理教室
WindowsNT	10201 ~ 10303,51	情報処理教室
X ウインドウ環境	10201 ~ 10303,21	情報処理教室

- MacOS からの利用

Apple Menu から Netscape を選択します。Apple Menu は画面左上にあるリンゴマークを押すと出てくるメニューのことです。

- WindowsNT からの利用

Internet Explorer のアイコンをダブルクリックします。

- X ウインドウ環境 からの利用

プロンプトから netscape と入力し <Return>を押します。

Ultra1 の場合

```
cssol001(80)% netscape &
```

もしここで

```
Error: Can't open display:
```

のようなメッセージが表示された場合は何かを間違えています。87 ページの 4.5.2、特に「非 X ウインドウ環境」での起動方法とエラー対策を参考にしてください。

## 5.6.5 ブラウザの終了

### Netscape の場合

ファイルメニューから "終了" を選択するとブラウザが終了します。また、ファイルメニューから "閉じる" を選択すると今使っているブラウザのウインドウだけが終了します。

### Internet Explorer の場合

ファイルメニューから "閉じる" を選択すると今使っているブラウザのウインドウを終了します。ウインドウが複数開かれている時は、その枚数分 "閉じる" を行います。

## 5.6.6 ブラウザの使い方

ここでは、Netscape のバージョン 4 (日本語版) と Internet Explorer のバージョン 3 (以後 IE3) を対象に説明していきます。

もし、それぞれのブラウザでバージョンが違っている場合があっても、使い方の基本はあまりかわっていないのでここでの操作を参考にしてください。

まず Netscape のメニュー画面を説明します。

メニュー画面を見ると次の項目があります。

戻る	直前の WWW ページに戻ります
次	「戻る」を使った場合に、戻った分を進めます。
再読み込み	今表示している WWW ページをもう一度読み込みます。
ホーム	ブラウザに設定されているホームページを表示します。 (情報処理教室のコンピュータでは京都産業大学のホームページが表示されます)
検索	ネットスケープ社が運営している検索ページに移動します。(お勧めしません)
ガイド	ネットスケープ社が運営している解説ページに移動します。(お勧めしません)
印刷	表示されている画面をプリンタに印刷します。
セキュリティ	ブラウザのセキュリティ情報を設定します。(通常は使いません)
停止	表示しようとしているページの読み込みを中止します。
ブックマーク	よく行く URL を登録しておきます。利用する時はこの一覧から行きたいところをクリックすると直接その URL のページを表示します。 (情報処理教室ではたくさんの人が共用するので使用しないでください)
場所	URL を入れる場所です。この欄に見たいページの URL を直接入力すると、そのページを表示します。



図 5.13 Netscape4.0 のメニュー画面

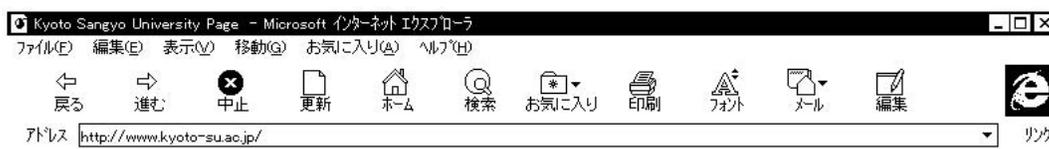


図 5.14 IE3 のメニュー画面

次に IE3 のメニュー画面を説明します。

戻る	直前の WWW ページに戻ります
進む	「戻る」を使った場合に、戻った分を進めます。
中止	表示しようとしているページの読み込みを中止します。
更新	今表示している WWW ページをもう一度読み込みます。
ホーム	ブラウザに設定されているホームページを表示します。 (情報処理教室のコンピュータでは京都産業大学のホームページが表示されます)
検索	マイクロソフト社が運営している検索ページに移動します。(お勧めしません)
お気に入り	よく行く URL を登録しておきます。利用する時はここの一覧から行きたいところをクリックすると直接その URL のページを表示します。 (情報処理教室ではたくさんの方が共用するので使用しないでください)
印刷	表示されている画面をプリンタに印刷します。
フォント	表示フォントの大きさを変えます。大きさは最小、小、中、大、最大の4種類あり、「フォント」をクリックするごとに最小 小 中 大 最大 最小の順で大きさが変わります。
メール	メールの送受信をします。(情報処理教室では使用しないでください)
アドレス	URL を入れる場所です。この欄に見たいページの URL を直接入力すると、そのページを表示します。

### 5.6.7 調べたいキーワードがある場合のブラウザの使用例

おいしいオムライスの作り方を紹介しているページを捜したい、釣りの穴場を紹介しているページを捜したいなど、あるキーワードがあって、それについて書かれたページを捜したい時の使い方です。

ここでは、試しに goo の検索ページに移動し、京都産業大学の情報を検索してみます。京都産業大学のホームページからリンクを使って、goo に移動します。

- ブラウザの「ホーム」をクリックして産業大学のホームページを表示します。そして、画面の中の「情報サービス」をクリックします。
- 「情報サービス」のページから「検索ページ集」をクリックします。
- 移動先のページには検索ページの一覧があります。この中に「goo」という項目がありますから、ここをクリックします。
- ここまでで、goo のホームページにたどりつけるはずですが、無事、goo のホームページにたどりつけたら今度はここで検索をしてみましょう。ここでは kyoto-su と university の2つをキーワードに検索してみます<sup>45</sup>。

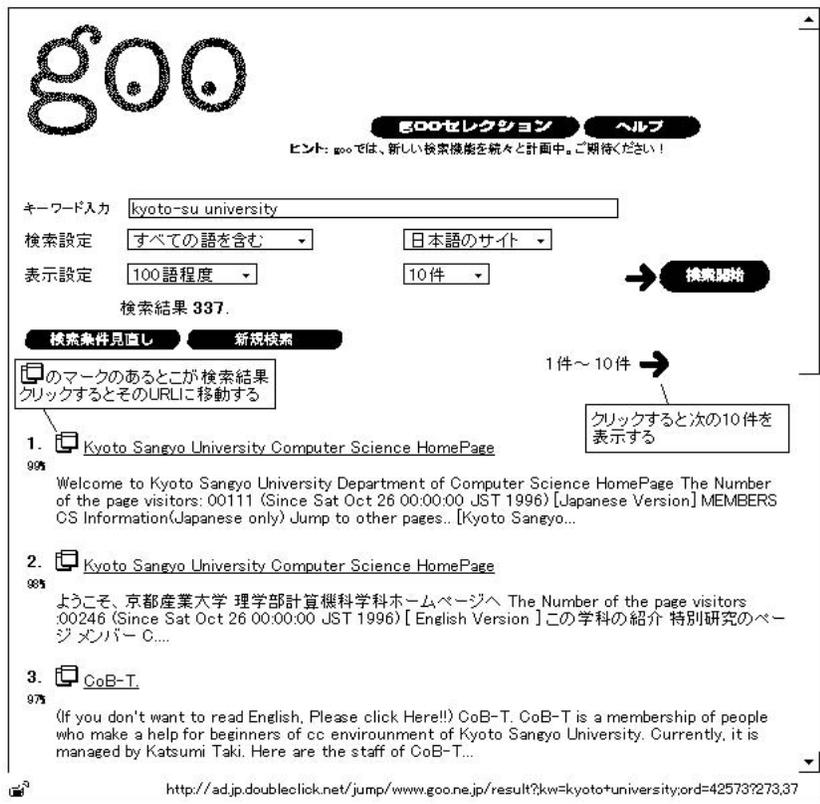
<sup>45</sup> もちろん「京都産業大学」とストレートにキーワードを指定したらいいのですがそれではあまり面白くありません。(^-;) )

検索のやり方は簡単です。kyoto-su と university の 2 語を間にスペースを入れて「キーワード入力」の右側の空白に書き込み「検索開始」をクリックします。

The screenshot shows the Google search page with the following elements:

- goo** logo
- Buttons: **gooセレクション** and **ヘルプ**
- Hint: ヒント: よりダイレクトな検索のためには「語種を保持」検索(でも可)が有効です!
- Keyword input:
- Search settings: **検索設定** (すべて of the 語を含む), **日本語 of the サイト**
- Display settings: **表示設定** (100 語程度), **10 件**
- Buttons: **goo の使い方** and **→ 検索開始**
- 追加機能** section:
  - **追加検索**: さらに語句を絞り込み追加検索ができます。
  - **日付検索**: ページが制作、更新された日時を指定し検索ができます。
  - **検索先設定**: 検索先をロケーションごとに指定し検索ができます。
  - **データタイプ**: データタイプを指定し検索ができます。

少し待っていると検索結果が表示されます。kyoto-su と university を含んだページの一覧が表示されていることでしょう。



- 検索結果をみる

検索結果がある場合は、ブラウザに「1.~」、「2.~」という感じで該当するページのタイトルが表示されます。目的のページのタイトルが検索結果にあれば、その行をクリックしてください。そのページに移ります。

- 他の検索結果を見る

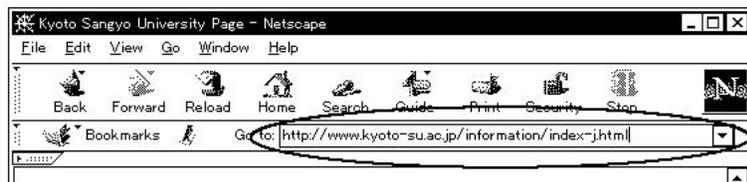
表示したページが思っていたものでなかった場合は、もう一度検索結果のページに戻り、別の結果を見ることになります。検索結果のページに戻るには、ブラウザの「戻る」をクリックしてください。その後で、先ほどとは別のタイトルをクリックしていけばいいでしょう。

後は、「戻る」、別のタイトルを選ぶを繰り返すことで検索結果を見ていくことができます。

### 5.6.8 URL を知っている時のブラウザの使用例

雑誌やテレビで紹介されていたりして、見たいページの URL が既に判っているときはそのページを直接表示できます。例えば見たいページのアドレスが <http://www.kyoto-su.ac.jp/information/index-j.html> だったとします。これをブラウザの「場所」(MSIE では「アドレス」)と書かれている場所の右側のボックスに直接入力するだけです。入力するにはボックスをクリックしてください。

もしかすると、URL を入力している時に入力していない文字が勝手に現れることがあるかもしれませんが、これはブラウザの機能なのですが、もしそのようなことが起きても気にせず、正しく URL を入力すれば OK です。



## 5.6.9 情報送信

ブラウザを使って WWW を利用する場合は、基本的に情報を受けるだけになりますが、場合によってはこちらから情報を送信する場合があります。これは大きく分けて 2 種類あり、電子メールを使って送信をする場合と、ページに情報を入力して送信ボタンを押す場合とがあります。

### 電子メールを送る場合

電子メールを送る形の場合は、ページの中にリンク機能を使って電子メールが送れるようになっています。そのリンク部分をクリックすると送信メールを作成する画面が開き、メールを書けるようになります。但し、これはブラウザがその機能に対応している場合だけです。また、当たり前の話ですが、メールアドレスを取得していないとメールを送ることはできません。それなら、その条件に合えばメールを送っても良いのかということになりますが、これができるブラウザでも公共利用の情報処理教室からこの方法でメールを送ることに問題があります。どういう問題があるかについては後で詳しく説明しますが、情報処理教室では相手のメールアドレスを確認して、後で mule 等でメールを送るようにしましょう。相手のメールアドレスは、そのページの中で

```
mailto:ozaki@cc.kyoto-su.ac.jp  
E-Mail:ozaki@cc.kyoto-su.ac.jp
```

と書かれていると思います。この場合は ozaki@cc.kyoto-su.ac.jp がメールアドレスですね。このメールアドレスを控えておき、後で mule 等を使ってメールを送るようにしてください。

### 情報を入力して送信する場合

もうひとつの情報送信は、情報を書込むボックスがページの中にあり、そこに必要事項を書込んで送信ボタンを押すというものです。こちらの場合は、アンケートを送ったり、会員になる場合によく使われる方法です。この場合はメールアドレスが無くても情報を送れるのが利点です。ただし、この場合もプライベートな情報を書く場合があるので注意が必要です。無闇に住所や電話番号等のプライベート情報を書き込むと、第三者に知られる恐れがあることを理解してください。信用のできないページに自分の情報を登録するのは止めるようにしましょう。

### ページ上でのパスワードについて

情報を入力して送信する場合について、もうひとつ注意することがあります。ページによっては会員登録をするためにユーザ ID とパスワードの入力を求めてくるものがあります。例えば、就職関係で有名なリクルート社のページにリクルートナビがありますが、このサービスを利用するためには会員登録が必要で、会員登録のためにはユーザ ID とパスワードの入力が必要です。

ここで使うユーザ ID とパスワードは、学内で使うユーザ ID、パスワードとは関係ありませんので絶対に違うものにしてください。学内で使うユーザ ID、パスワードは学内のコンピュータを使って、cc 環境に接続する時だけ使います。もし、WWW 上であなたのユーザ ID、パスワードが入力された場合は、あなたの学校でのパスワードを人に教えたこととなります。

会員登録をしている場合は、おそらくメールアドレスも書き込んでいるでしょう。ここでもし、学内の cc 環境で使っているユーザ ID、パスワードを入力してしまったら、あなたのメールがすべて読まれたり、もっとひどいことをされる危険性があります。

絶対に、WWW 上の会員登録では cc 環境で使っているユーザ ID、パスワードを入れないようにしましょう。もし、入力してしまった時は、できるだけ早く学校のパスワードを変更してください。

## 日本語の入力

ブラウザに対して日本語を入力する場合、パソコンで使っている場合はパソコンの日本語入力機能を使ってください。cc2000 などで X ウィンドウ環境で使っている場合は面倒ですが、一度 Mule で文字を書き、それをコピーしてください。マウスを使えば Mule 側でコピーしたい部分を左ボタンでドラッグして、ブラウザ側の書き込みたい部分で真中のボタンを押すとペーストできます。2 つしかボタンのないマウスなら左右のボタンを同時に押すことで真中ボタンの代用ができるでしょう。

## ブラウザの問題点

いくつかのブラウザは WWW のブラウザとしてだけでなく、mail や news の読み書きもできます。

しかし、それらはユーザ ID に対してパスワードを確認しないので、自分のメールアドレスを間違えて登録してしまってもそのままメールが送れたりニュースを投稿することができてしまいます。

これは大変困ったことです。この間違えたメールアドレスでメールやニュースを送った場合は、読んだ人が送った人にメールを返そうとしても、その人にメールが届かなくなります。返信を書いた人がとても重要なメールを書いていたとしても、それが相手に届かないということになるのです。

また、その間違えたメールアドレスがたまたま別の人のアドレスになってしまった場合は、どうなるでしょうか？この場合は送った人に対する返信がすべて、その別の人のところに届くことになります。そうすると間違えられた人は、身に覚えの無いメールやニュースの返信を受け取ることになります。

返信を書いた人にもアドレスを間違えられた人にも迷惑がかかることになりますね。悪意を持ってメールアドレスを間違えたとしたら尚更です。

本当ならこういう危険性のあるブラウザは使うべきではないでしょう。しかし他に安全で使い勝手の良いブラウザがないのも確かです。そこで、使う人間がその危険性を理解して危険のない状態で使うようにしないといけないのが現状です。

利用者の皆さんは、ブラウザの長所、短所を良く理解して使うようにしてください。メールを送るとか、ニュースを投稿するという場合は Mule を使うようにしましょう。

### 5.6.10 トラブルを出さないために

WWW のブラウザで mail や news を使うことの問題点は理解しましたね。この問題点を具体的に回避するために、「共用パソコンのブラウザで mail や news の利用を禁止する」ことが提案されました。しかし様々な事情から現状ではそれを禁止しないことにしました。

禁止とはならなかったものの、その話し合いの中でずいぶん収穫がありました。それは「パソコンの WWW ブラウザを mail や news に使うことの問題点がどのようなものか知っている」、さらに「その様な

問題点を排除することは大切だ」という認識をもつ方が大部分だということです。これを無駄にしたいことはありません。「トラブルは出さないという姿勢」こそが大切なわけです。この姿勢さえあれば、かなりの割合で問題の発生を回避できると考えられるからです。

京都産業大学からアドレスに関するトラブルは出さないようにしましょう！

これは大学としての姿勢です。

危ない使い方だと知らないで使っている人がいれば、安全な方法があることを教えてあげればいいのです。全く分からない人は、知っていそうな人に尋ねればいいのです。尋ねることは決して恥ずかしいことではありません。そうして、どうすることがより良い方法なのか皆で考えましょう。

### 5.6.11 あなたも作れるホームページ

皆さんはインターネットのサービスをフル活用できる環境にいるのです。WWW を見てだけでなく、自分のホームページを作ってみませんか？本学では学生が自由にホームページを作れる環境があります。利用するための条件は

- 本学の学生で、ユーザ ID の交付を受けていること。
- 法律を守ること。

これだけです。法律とは著作権や肖像権等です。WWW のページは新聞に記事を掲載するようなものです。そのまま新聞に載せたとして問題がないかどうかを考えてください。

詳しくは、HTML の章を参照してください。

### 5.6.12 FTP 再び

146 ページ、5.5 で ftp サーバからソフトウェアをコピーする方法について述べました。実は anonymous FTP に限り、ブラウザを使ってコピーすることが可能です。ここでは Windows 版の Netscape Navigator バージョン 4 を例に説明をします。使っているブラウザが、Netscape Navigator バージョン 4 でない場合も基本操作は同じですので、ここを参考にしてください。また、Macintosh を使うときはマウスボタンが一つしかありませんので、クリック時の操作が若干違います。右クリックとあるところでは、マウスボタンを数秒押しつづけるような操作をしてください。

ではブラウザを使ったソフトウェアのコピーの具体例として Windows95 対応の telnet ソフト、TeratermPro を入手する方法について説明していきます。

#### ftp サーバへのアクセス

まず ftp サーバにブラウザを使ってアクセスします。アクセスは、URL に ftp サーバを指定するだけで簡単に行うことができます。

WWW アクセスの時には URL に `http://www.kyoto-su.ac.jp/` として WWW サーバにアクセスしていました。ftp サーバへのアクセスもこれと同じように行います。但し、今まで `http` と書いていたところに `ftp` と書きます。

具体的には、産大の ftp サーバへのアクセスは URL に

```
ftp://ftp.kyoto-su.ac.jp/
```

と書きます。



## ディレクトリ移動

ftp サーバにアクセスするとディレクトリの一覧が表示されます。今回の目的である teraterm は /pub/Windows/NetTools/Telnet/teraterm 以下にありますので、目的の場所までディレクトリをクリックして目的の場所に移動していきます。順に pub/、Windows/、NetTools/、Telnet/、teraterm/ とクリックすればいいでしょう。

## ファイルのコピー

teraterm のディレクトリまでくれば、後は telnet をコピーするだけです。

コピーの仕方は、目的とするソフトウェアをマウスの右ボタンでクリックし、下から 2 番目にある「リンクを名前を付けて保存」を左ボタンで選びます。そうすると保存する名前と場所を聞いてきますから、任意の名前を付けるとコピーを開始します。

Mac ユーザの場合は、右ボタンを押す代わりにマウスを長く押しつづけます。そうすると、Windows 版と同じような感じでメニューが出てきますからそこで保存を選んでください。

## 第6章 UNIXもっともっと

ここでは計算機センターが管理している UNIX 環境を例に取りながら、より進んだ UNIX 環境の利用方法の説明と、さまざまな機能の紹介をします。ここでの説明は網羅的なものではなく、部分を取り上げて曖昧に説明しています。これは本文の読者のコンピュータそのものについての知識のハードルを高くしすぎないようにしているためです。読者が UNIX、つまりコンピュータの利用に慣れて行くにしたがって自分でマニュアル、書籍、ネットワークなどから情報を常に取り込んで理解することが大切です。

### 6.1 ファイルの整理

cc 環境ではファイルは UNIX コンピュータのディスクと呼ばれる装置に保存されています。このディスクと呼ばれる装置に格納できるファイルの量には限界が有り、cc 環境で利用できるディスク装置の数も限られています。この限られた資源を千人を超える利用者で共同利用しているのですから、おのずと一人一人が利用できるディスクの量も限られてきます。これはすなわち利用者ができることのできるファイルの数やその量の合計には限りがあるということです。その限界を超えてファイルを作ろうとしてもエラーメッセージが表示され、ファイルを作成することができません。

cc 環境では利用者それぞれによるファイルの整理が必要なのです。

ディスクを大量に使っている人は login したときに「ディスクを既定値を超えて利用しているので不要なファイルを消去せよ」という意味のメッセージを見ることがあるかもしれません。これはその人が、一人の利用者が使用する事のできるディスクの上限値を超えている事を意味しています。

このメッセージが表示された場合は、とにかくにも不要なファイルを消去しないといけません。

#### 6.1.1 ファイルの大きさを調べる

ファイルの整理をするには、どれでも不要なファイルを削除すれば良いのですが、とにかく闇雲に削除するのではなく、サイズの大きなファイルから削除する方が良いでしょう。ファイルのサイズを見る方法は幾つかありますが、ここでは二つ紹介します。

ls -l : 個々のファイルの大きさを調べる

ls コマンドに引数として -l オプションを加えれば、ファイルの大きさが判ります。ファイルの大きさは Byte、バイトと呼ばれる単位で表現され、ls -l で表示される情報のうち、左から 4 番目にあるものです。例を挙げます。

```
cc2000(93)% ls -l core
-rw-r--r--  1 root    daemon   153248 02月 22日 22:32 core
cc2000(94)%
```

上記の例では core というファイルの大きさは 153248 バイトであることが判りました。ファイルの種類によって一概には言えないのですが、単純には 1 文字は 1 バイトで表現できますので、153248 文字くらいの内容がこのファイルには含まれていると考えられます<sup>1</sup>。

ls コマンドはファイル名の指定をせずに、単に ls -l<Return>として実行した場合はカレントディレクトリのファイル全ての情報を表示します。

しかしカレントディレクトリに存在するディレクトリファイルについては、そのディレクトリの下にあるファイルの量をまとめて表示してくれるわけではありません<sup>2</sup>。ディレクトリファイル以下にあるファイルの大きさについては、そのディレクトリ以下に cd コマンドで移動して、もう一度 ls -l<Return>とするしかありません。それが面倒ならば ls -lR<Return>で階層的にファイルの情報を表示させることもできますが、これはかなり見にくいものです。

### du -s : ディレクトリ以下に含まれるファイル大きさの合計を調べる

ls -l コマンドではディレクトリ以下に含まれるファイルの容量の合計を知ることは簡単にはできません。そのような場合は du コマンドに -s オプションを付けて実行します。以下に Mail ディレクトリ以下のファイルの量の合計を表示する例を示します。

```
cc2000(82)% du -sk Mail
39      Mail
cc2000(83)%
```

勿論ファイル名に Mail ではなく \* を指定すればカレントディレクトリ以下の全てのファイル<sup>3</sup>の量を計算した結果が表示されます。

注意が必要なのは du コマンドは、その結果をバイト単位ではなくブロック単位、もしくはキロバイト単位で表示することです。1 Block = 512 Bytes , 1 KiloBytes = 1000 Bytes (場合によっては 1 KiloBytes = 1024 Bytes) となります。一般的にはキロバイト単位の方が扱い易いと思われるので、du コマンドを利用するときには -sk というように表示をキロバイト単位で行うようなオプションを付けておくといいでしょう。

上記の例では Mail というディレクトリ以下のファイルの量の合計は 39 キロバイトという事を意味しています。

---

<sup>1</sup> この解釈は余りにも単純すぎて、現実には妥当ではありませんね。計算機の内部構造に興味のある人はもう少し自分で調べて納得してください。ここでは説明しません。

<sup>2</sup> このディレクトリファイルの「大きさ」については無視してください。これまた計算機の内部構造に興味のある人は調べてみるといいでしょう。

<sup>3</sup> 厳密には名前が . で始まるファイルについては表示されません。

### 6.1.2 身に覚えのないファイル

UNIX を利用していくと、様々なファイルを作って作業することになるでしょう。それらのうち、幾つかはあなたが作った覚えのないファイルもあるでしょう。

以下にそれらのうち代表的なものを示します。

**Mail** 電子メールを読むためにあらかじめ作られているディレクトリです。削除しない方がいいでしょう。

**Wnn6** 漢字変換をするためにあらかじめ作られているディレクトリです。削除しない方がいいでしょう。

**Apps, Library** cc 環境の古いコンピュータ (NeXTStation) のためのディレクトリです。必要なければ削除して構いません。

**core** プログラムやコマンドが実行している途中で何等かの原因によって実行が中断されたときに強制的に作成されます。このファイルの意味、使い道を知らない人は無条件に削除して構わないでしょう。むしろ大きさが結構大きいものが多いので積極的に削除すべきです。

**dead.letter** メールを書いている最中に何等かの原因によって作業が中断したときに強制的に作成されます。中身はきっと最後に書き掛けていたメールの中途半端な中身だと思います。内容を `more dead.letter<Return>` などして確認した後、保存の必要がなければ削除すれば良いでしょう。

**mbox** メールを扱うには MHE だけでなく、`mail` というコマンドも利用できます。このファイルは `mail` コマンドが作成するものです。cc 環境では `mail` コマンドによる日本語メールの読み書きは、その動作を保証していません。MHE 環境への移行を薦めます。移行が済めば `mbox` ファイルは内容を `more mbox<Return>` などして確認した後、保存の必要がなければ削除すれば良いでしょう。

名前は固定ではありませんが、ファイルの名前に特徴のあるものが勝手にできている場合があります。以下に代表的なものを示します。

#### #abc# など、# で囲まれたファイル

Mule がファイルを修正したときに一時的に作成するファイルです。Mule の作業が何等かの理由で中断した為に残ったものと思われます。内容を確認した後、保存の必要がなければ削除すれば良いでしょう。

#### abc~ など、~ が末尾に付くファイル

Mule がファイルを修正したときに、修正前のファイルの内容をそのまま「~」を付けた名前のファイルに残します。例えば `abc` というファイルを修正し、保存すれば、修正した後の内容は `abc` ファイルに残り、修正する前の内容は `abc~` ファイルに残るということです。内容を確認した後、保存の必要がなければ削除すれば良いでしょう。

#### .cshrc , .login など、. が先頭に付くファイル

これらのファイルは大抵の場合何等かの環境設定ファイルです。内容が判らない場合は不用意に変更したり削除したりしない方がいいでしょう。なお、これらの先頭が「.」(ピリオド)で始まる名前のファイルは普通に `ls` コマンドなどを実行してもその一覧には表示されません。ある意味では「隠された」ファイルとなっているのです。これらのファイルの一覧を表示させるには `ls -a<Return>` とするのが良いでしょう。

これら以外にも様々なファイルが知らない間にできているかもしれません。小さなファイルはまだ良いのですが、大きなものは注意して、不必要と思ったら削除しておく方がいいでしょう。どうしても何故できた

のか判らないようなファイルができるのであれば計算機センターのメインカウンターか、sandai.question ニュースグループに相談してください。

### 6.1.3 ファイルを削除する

ファイルの削除には `rm` コマンドを利用します。 `rm` ファイル名<Return>です。ディレクトリを削除するには `rmdir` コマンドを利用します。 `rmdir` ディレクトリ名<Return>です。ディレクトリ以下にファイルが残っていて、それらのファイルごと削除する場合は `rm -r` ディレクトリ名<Return>とします。

### 6.1.4 ディスクの利用量をチェックする

ファイルの量が限界に到達してから仕方なくファイルを削除するよりは普段からディスクの使用量をチェックしておく方がいいでしょう。あなたが現在使っているファイルが占めているディスクの中での量を表示するには `quota -v` コマンドを利用します。以下に例を示します。

```
cc2000(88)% quota -v
Disk quotas for sheryl (uid 2019):
Filesystem      usage  quota  limit   timeleft  files  quota  limit   timeleft
/NF/home         1433  30000  30000           141  10000  10000
cc2000(89)%
```

ここで表示される内容の意味を左から順に説明します。

---

Filesystem	余りにしないでください。(^^;)
usage	今使っているファイルの量の合計を表します。例では 1433KiloBytes 使っていることとなります。
quota	利用できるファイルの量の上限を表します。例では 30,000KiloBytes、つまり 30MegaBytes を意味しています。
limit	直前の quota と同じです。違いは気にしないでください。(^^;)
timeleft	余りにしないでください。(^^;)
files	今使っているファイルの数の合計を表します。例では 141 ファイルあることとなります。
quota	利用できるファイルの数の上限を表します。例では 10000 ファイルを意味しています。
limit	直前の quota と同じです。違いは気にしないでください。(^^;)
timeleft	余りにしないでください。(^^;)

---

すなわちファイルについてはその量と個数のそれぞれの合計に独立に制限があるのです。しかし現実には個数が限界を超えることは殆ど無いと思われまので、要は左から 2 番目の `usage` が 3 番目の `quota` を超えないように調整していれば良いわけです。

更に細かくディレクトリ毎に使用量を調べたければ `du -sk *`とすればいいでしょう。

## 6.2 情報検索

### 6.2.1 学内の利用者情報の検索

電子メールやネットニュースなどを利用していると、あの人のユーザ ID は何だったかな?と思うときがあるでしょう。の4「UNIX それから」では finger コマンドによる利用者情報の取得方法を紹介しましたが、ここではさらに詳しく利用者の情報を得ることができる whois コマンドを紹介します。

但し注意が必要なのはここで紹介している whois サービスは主として cc 環境の為に提供されているもので、全ての学内の UNIX コンピュータで利用でき、全ての学内の UNIX コンピュータ利用者の情報が登録されているわけではないということです。cc マシン群ではこのサービスが利用できますが、他の UNIX コンピュータからどの程度利用できるかは保証できません<sup>4</sup>。

whois コマンドを利用すると利用者のユーザ ID などが検索できます。whois keyword<Return>です。以下にキーワードとして sigeta hiroyuki を与えた場合を示します。キーワードには空白が含まれて居ますので、念のため引用符でかこんで与えています。

```
cc2000(81)% whois 'sigeta hiroyuki'
[ Whois service for Kyoto Sangyo University.      1994.7.7 serviced from ccnic.]
[ Please contact ozaki@cc.kyoto-su.ac.jp if you find some trouble.                ]
[ For more information, use 'whois -h ccnic.kyoto-su.ac.jp help:'                ]
```

```
ユーザ ID,uid : sheryl , 2019
氏名          : Sigeta Hiroyuki
              : 重田 裕之
所属          : 計算機センター事務室 (事務職員)
アカウント    : sheryl@cc sheryl@cs
cc2000(82)%
```

教員、学生など、その人の種別によって表示される項目は様々です。ユーザ ID の横に表示されている uid は個人を識別するためにユーザ ID と一対一に割り当てられている番号です。whois コマンドのキーワードとしても使えます。例えば上記の例では whois 2019<Return>としても同じ結果が得られるでしょう。最後にアカウントという情報が表示される場合があります。これはその人がそこに表示されているシステムにユーザ登録されている事を示しています。しかし登録されているからと言って、そのアドレスにメールを送れば必ず読んでくれるとは限りません。あくまでメールを送るときの目安程度に考えてください。

キーワードにはユーザ ID、名前、uid、所属などが使えます。また、これらの文字列の一部でも与えることができます。先ほどの例ではキーワードにローマ字のフルネームを用いましたが、この一部分として whois hiroyuki<Return>などのようにする事もできます。以下に例を示します。但し hiroyuki というキーワードにマッチする利用者はちょっと多かったようです。このような場合はマッチした人の一覧を挙げるようになっていきますので、ここから uid などを元にもう一度 whois を行い、目標の人の詳細情報を得ることができます。

```
cc2000(82)% whois hiroyuki
[ Whois service for Kyoto Sangyo University.      1994.7.7 serviced from ccnic.]
[ Please contact ozaki@cc.kyoto-su.ac.jp if you find some trouble.                ]
```

<sup>4</sup> 因みに cc 環境 whois サービスは ccnic マシンが提供しています。cc 環境マシン群以外から利用する場合は whois -h ccnic keyword などとしてください。

```
[ For more information, use 'whois -h ccnic.kyoto-su.ac.jp help:' ]
```

uid	ユーザ ID	種別	名前
1913	hi44kubo	K	Kubo Hiroyuki 久保 宏之
2019	sheryl	S	Sigeta Hiroyuki 重田 裕之
2327	hiro	K	Itou Hiroyuki 伊藤 浩之
10265	gamera	G	Kamemoto Hiroyuki 亀本 裕之
	:	途中略	:
13559	hkodama	G	Kodama Hiroyuki 児玉 寛将
19026	oonishi	G	Oonisi Hiroyuki 大西 洋幸

種別 = G:学生 I:院生 K:教員 S:職員 M:その他

```
Many machting. See the detail, use uid number for key.  
cc2000(83)%
```

ただ、余りにも漠然としたキーワードを与えた場合はマッチする利用者が多すぎ、以下のようなメッセージを表示して情報を表示しない場合があります。

```
cc2000(81)% whois hide  
[ Whois service for Kyoto Sangyo University.      1994.7.7 serviced from ccnic.]  
[ Please contact ozaki@cc.kyoto-su.ac.jp if you find some trouble.                ]  
[ For more information, use 'whois -h ccnic.kyoto-su.ac.jp help:' ]
```

```
Too Many machting. Try again with more detail information.  
cc2000(82)%
```

このような場合はキーワードを hide だけではなく、hideyuki などにするなどしてもう少し条件を絞り込んで whois を再実行してください。または以下のような指定も可能です。この例では山田だろーさんや山田たけしさんがマッチします。

```
whois 'part:yamada ta'
```

## 6.2.2 学外の組織などの情報を調べる

6.2.1 で紹介した whois コマンドは cc 環境で独自に提供されているものです。学内の他の UNIX コンピュータで同じようにこのコマンドが通用するとは限りません。

産業大学以外の組織でも whois サービスが行なわれている場合があります。ここではそれらのうち、Internet を通じて公開されている whois サービスの使い方を紹介します。以下に紹介する公開 whois サービスは主にネットワーク関係の情報を提供しています。サービスは具体的には Internet 上のあるマシンで行なわれており、そのマシンに目がけて whois コマンドでキーワードの問い合わせを行なうのです。

whois.nic.ad.jp 国内のネットワーク関係の情報を提供しています。これは JPNIC（日本ネットワークインフォメーションセンター）という組織が行なっているサービスです。

rs.internic.net Internet 関係の情報を提供しています。これは Network Solutions Inc. というアメリカにある組織が行なっているサービスです。

nic.ddn.mil MILNET 関係の情報を提供しています。MILNET はアメリカの軍事関係の組織が繋がれたネットワークです。ですからこのサービスは国防省が行なっているはずです。(ちょっと自信がありませんが。)

### 組織の情報を得る

例えば京都産業大学のドメイン名 (Internet 上の組織もしくは組織のネットワークの名前) を知りたい場合は、国内の情報ですから whois.nic.ad.jp に問い合わせます。このホスト名は whois コマンドに -h に続いてオプションとして与えます。whois -h whois.nic.ad.jp. 'kyoto sangyo'<Return>と入力してください。キーワードに空白が含まれるので念の為にキーワードを引用符で囲んでいることに注意してください。以下に例を示します。

```
cc2000(86)% whois -h whois.nic.ad.jp. 'kyoto sangyo'
[ JPNIC database provides information on network administration. Its use is ]
[ restricted to network administration purposes. For further information, use ]
[ 'whois -h whois.nic.ad.jp help'. To suppress Japanese output, add '/e' at ]
[ the end of command, e.g. 'whois -h whois.nic.ad.jp xxx/e'. ]
```

```
京都産業大学 (Kyoto Sangyo University)          KYOTO-SU.AC.JP
京都産業大学 (Kyoto Sangyo University)          133.101.0.0
```

To single out one record, look it up with '!xxx',  
where xxx is a key shown on the right hand side.

```
cc2000(87)%
```

京都産業大学のドメイン名 kyoto-su.ac.jp とネットワークアドレス 133.101.0.0 が表示されました。詳細な情報を得る為に今度は キーワードに kyoto-su.ac.jp を与えて検索してみましょう。例は挙げません。試してみるといいでしょう。逆にドメイン名からその組織についての情報を得ることもできます。

勿論海外の組織やドメイン名などについての情報を得る為には whois -h に続けるホスト名を rs.internic.net. などに置き換えれば大丈夫です。

### 人の情報を得る

今度は例えば Internet 上の sherylさんを調べてみましょう。

whois -h rs.internic.net. sheryl<Return>と入力してください。以下に例を挙げます。

```
cc2000(93)% whois -h rc.internic.net. sheryl
whois: rc.internic.net.: cc2000(139)% whois -h rs.internic.net. sheryl
Birely, Diane (DB1506)          sheryl@SPRINGACRE.COM          714-851-8356 x.256
Bruno, Sheryl (SB657)          sheryl@COMMUNITY.NET          707-427-8458
Coppenger, Sheryl (SC295)      sheryl@SEAS.GWU.EDU          (202) 994-6853
Galchutt, Sheryl (SG420)      sheryl@NOC.DIGEX.NET          (703) 751-1905
Gruenler, Sheryl (SG558)      Sheryl@SPRINGACRE.COM          714-857-0300
```

Hamlin, Sheryl (SH195)	sheryl@DIGIRATI.COM	415-543-8290
Hungerford, Sheryl (SH956)	Sheryl@ADAGIOSYS.COM	(619)929-9565
Lindsell-Roberts, Sheryl (SL712)	sheryl@TIAC.NET	508-229-8209
Sheryl Linsell_roberts (SHERYLWRIT-DOM)		SHERYLWRIT.COM
Sheryl, Campbell (CS406)	postmaster@ENTHONE-OMI.COM	203-799-4922

The InterNIC Registration Services Host contains ONLY Internet Information (Networks, ASN's, Domains, and POC's).

Please use the whois server at nic.ddn.mil for MILNET Information.

cc2000(95)%

組織やら人やらの情報がごちゃまぜに表示されています。目当ての sheryl こと'sigeta hiroyuki' は残念ながらありませんでした。ひょっとして表示されると思っていました? Internic 上に登録されている人というのは一般的にはネットワーク管理者である事が多いようです。それ以外の人はあまり登録されていない感じですが。しかし逆にネットワーク上での有名人というのは割合にネットワーク管理者だったりしますから、結構役に立つのです。

人の情報と違って InterNIC ドメインの情報についてはほとんど完全に登録されているはずですが。

## 6.3 ファイルのアクセス権

誰か友達がファイルを作っていて、そのファイルをあなたが自分のホームディレクトリ以下にコピーしたいと思ったとします<sup>5</sup>。ファイルの階層構造を理解したあなたは早速以下のようなコマンドでコピーをしようとするでしょう。

```
cc2000(85)% cp ~tanaka/sample.tex ~
cp: /NF/home/g840/tanaka/sample.tex: Permission denied
cc2000(86)%
```

多くの場合は上の例のように失敗してしまいます。このエラーメッセージ「Permission denied」はUNIXを使っていると、時々みかけるエラーですね。これは「あなたにはアクセス権が無いよ」ということを意味しています。

### 6.3.1 アクセス権

UNIX コンピュータはみんなで使うコンピュータであることは初めに説明しました。そのため、あなたのホームディレクトリのとおりには他人のホームディレクトリがあったりします。つまり誰でも他人のファイルの置き場所が大体わかっているわけです。これでは誰でも他人のファイルを覗いたり、書き込んだり、消去したりできてしまいます。

これは安全上（セキュリティ）の問題です。たった一人の人が専有して一台のコンピュータを使用し続けるのなら、そのコンピュータを自分の部屋に置いて、その部屋の入口に鍵を掛けておけば良いのです。でも cc 環境の UNIX のようにみんなで使うコンピュータは閉じ込めるわけにはいきませんからセキュリティを確保するためには何か別の対策が必要です。

そこでみんなで使うコンピュータの多くではアクセス権限という考え方を採り入れています。つまりコンピュータ上の資源にはそれぞれ誰が使えるかという情報が書いてあるのです。逆にその情報から洩れた人は使えないということです。コンピュータ関係の世界では、この「(資源を)使う」という事を「アクセスする」と表現することがあります。「アクセス権」とはつまり(資源に対する)「使用権限」という意味です。

そして、cc 環境では安全のため（もしくはプライバシーのため）に「利用者のホームディレクトリ以下のファイルは、利用者自身しかアクセスできない」という設定になっています。先の「Permission denied」はこの制限に引っかかったためのエラーメッセージだったのです。

### 6.3.2 UNIX におけるアクセス権

UNIX では全てのファイルに常にアクセス権が設定されています。ファイルが利用者によって作成されると、利用者がまずファイルの所有者となり、自動的にアクセス権が設定されます。アクセス権の設定は後で所有者によっていくらかでも変更することができます。

アクセス権は具体的には次の 3 種類の項目に対してそれぞれ許可を与えることによって設定します。

---

<sup>5</sup> もちろんその友達の了解を得て、ですよ！勝手に人のファイルをコピーしてはいけません。

種類 (略号)	ファイルの種類	設定することによって許可されるアクセスの内容
読み出し (r)	ファイル	そのファイルの中身を取り出す
	ディレクトリ	そのディレクトリ以下のファイルの一覧を表示する
書き込み (w)	ファイル	そのファイルの中身を書き変える、もしくは消去する
	ディレクトリ	そのディレクトリ以下にファイルまたはディレクトリを新たに作る、もしくはそのディレクトリを消去する
実行 (x)	ファイル	そのファイルを実行する
	ディレクトリ	そのディレクトリ以下に cd コマンドで移動する

上記の 3 種類のアクセス権の設定項目は「所有者 (user)」と「グループ (group)」と「その他の人 (other)」のそれぞれを対象に対して別個に割り当てられます。つまりあるファイルに対して「所有者は読み書きできて、グループは読むだけで、その他の人は読むことすらできない」という設定ができるのです。

所有者とはそのファイルを作った人です。グループについてはここでは説明しません。自分がどのグループに属しているかは id コマンドで確認することができますので、参考にしてください。学生は全員 student グループの一員です。教員は全員 teach グループの一員です。その他の人というのは所有者でもグループのメンバーでも無い利用者のことです。

### 6.3.3 アクセス権限を調べる

ls -lg コマンドで、ファイルのアクセス権限を含めた詳細な情報を得ることができます。書式は以下のとおりです。

```
ls -lg [ファイル名...]
```

以下に実行例を示します。

```
cc2000(82)% ls -lg
total 5
drwxr-xr-x  2 sheryl  clerk      512 Mar 14  1995 Apps/
drwxr-xr-x  2 sheryl  clerk      512 Aug 11  1995 Clang/
drwxr-xr-x  3 sheryl  clerk    3072 Oct  4  1994 Comics/
drwxr-xr-x  2 sheryl  clerk      512 Mar  7 17:23 ETC/
drwxr-xr-x  8 sheryl  clerk      512 Nov  2  1993 Library/
-rw-r--r--  1 sheryl  clerk      379 Mar  7 18:42 a.adr
```

```
cc2000(83)%
```

これ以降に以下の一行を取りだして、詳細に説明します。

```
-rw-r--r-- 1 sheryl clerk 379 Mar 7 18:42 a.adr
```

---

-rw-r--r--	ファイルの種類とアクセス権限についての情報。(さらに後述)
1	リンク数。ここでは説明しない。
sheryl	所有者
clerk	グループ
379	ファイルの大きさ。単位は Byte(バイト)。ここでは説明しない。
Mar 7 18:42	作成年月日
a.adr	ファイル名

---

この、ロングフォーマットで得られる出力の最初の部分 (-rw-r--r--) に注目してください。以下にそれぞれの桁について説明します。

桁位置 (例での値)	意味
1(-)	ファイルの種類を表す。d ならディレクトリ、- ならファイル。
2(r)	所有者に対する読みだし許可を表す。r なら許可、- なら禁止。
3(w)	所有者に対する書き込み許可を表す。w なら許可、- なら禁止。
4(-)	所有者に対する実行許可を表す。x なら許可、- なら禁止。
5(r)	グループに対する読みだし許可を表す。r なら許可、- なら禁止。
6(-)	グループに対する書き込み許可を表す。w なら許可、- なら禁止。
7(-)	グループに対する実行許可を表す。x なら許可、- なら禁止。
8(r)	その他に対する読みだし許可を表す。r なら許可、- なら禁止。
9(-)	その他に対する書き込み許可を表す。w なら許可、- なら禁止。
10(-)	その他に対する実行許可を表す。x なら許可、- なら禁止。

---

つまり以下のような表組を一行に引き延ばして書いたようなものです。

対象	読みだし	書き込み	実行
所有者	r	w	-
グループ	r	-	-
その他	r	-	-

---

### 6.3.4 アクセス権限を設定する

アクセス権限は所有者によって設定を変更することができます。chmod<sup>6</sup> コマンドを利用します。書式は以下のとおりです。

```
chmod mode file...
```

---

<sup>6</sup> change mode の略のつもりなのです

chmod コマンドは指定の file のアクセス権を mode の指定に従って変更します。mode は3つの部分からなる文字列で、対象... オペレータ 内容 となっています。以下にそれぞれの部分に与え得る記号とその意味を説明します。

対象	u	所有者
	g	グループ
	o	その他の人
	a	全ての人
オペレータ	+	追加
	-	取消
設定内容	r	読みだし
	w	書き込み
	x	実行

以下に chmod の具体例を挙げます。

1. `chmod a+r log` 全ての利用者に読みだし権限を与える
2. `chmod ug+rw log` 所有者とグループメンバーに読み出しと書き込み権限を与える
3. `chmod go-rw log` 自分以外の利用者から読み書きできないようにする

+ と- オペレータは元のアクセス権に新たな設定を「付加する」ように働きます。つまり元の設定で、影響を受けない部分はそのまま残ります。例えば 1. の例では書き込み権限や実行権限などには影響を与えず、もとの log ファイルが持っていたはずの書き込みに関するアクセス権限の設定はそのまま残ります。いろいろ試して、chmod の振舞いを理解するのがいいでしょう。

#### ちょっとマニアックな chmod の使い方

もっと直接的にアクセス権の設定をするために、chmod コマンドにはもう一つの mode 文字列の与え方があります。アクセス権限を表す `rw-r--r--` などの文字列を以下のようにして二進数に見立てて計算するのです。

```
rw-r--r--   rw-,r--,r--
110100100   110,100,100   6,4,4   644
```

つまり `rw` の部分で3つに分け、それぞれで二進数のつもりで計算するのです。r が  $2^2$  の桁、w が  $2^1$  の桁、x が  $2^0$  の桁、という訳です。計算すると、`rw-r--r--` は `rw-` , `r--` , `r--` と分割され、 $2^2 * 1 + 2^1 * 1 + 2^0 * 0$  ,  $2^2 * 1 + 2^1 * 0 + 2^0 * 0$  ,  $2^2 * 1 + 2^1 * 0 + 2^0 * 0$  となり、最終的に 6,4,4 となります。コマンドとしては `chmod 644 log` で完全に `rw-r--r--` を意味するアクセス権の設定ができます。

## 6.4 シェルよもう一度

### 6.4.1 ファイル名の補完

今までファイルを指定するときにはファイル名を全てタイピングしていたと思います。でも、ファイル名を全部タイプしなくてももっと簡単にファイルを指定することができる場合があります。その為にファイル名を記述する部分に用いる文字として\*と?があります。これらの文字をUNIXではメタキャラクタと呼んでいます。以下にそれぞれ詳しく説明します。

説明はカレントディレクトリ以下のファイルの状況が以下のようにだと仮定して行います。

```
cc2000(88)% ls
bb      cc      log      log2     sample
cc2000(89)%
```

任意の文字列に適合するメタキャラクタ「\*」

ここで「\*」文字を紹介しましょう。例えば上記の例でlogとlog2の両方のファイルの中身を見たい場合は、以下のようにすれば良いのです。

```
cc2000(92)% more l*
```

つまりl\*とは「lで始まり、それ以降はどんな文字列でも構わないファイル」を列挙することを意味しています。その結果more l\*はmore log log2と解釈されて実行されます。

\*が適用可能なのは何もファイル名の最後に限りません。

例えば上記の例でmore \*2とすればそれは「どんな文字列で始まって構わないから、最後が2で終わるファイル」と解釈され、log2 ファイルを指します。

\*o\*とすれば、それは「どんな文字列で始まって構わないから、途中にoを含み、またどんな文字列で終わっても構わないファイル」と解釈され、log log2 の二つのファイルを指します。

\*とすれば、それは「どんな文字列でも構わないファイル」と解釈され、bb cc log log2 sample の五つのファイルを指します。

\*文字はファイル名の記述のどこに幾つ現れても構いません。

また、適合する文字が無くても適合したとみなします。例えばlog\*とすれば「logで始まり、それ以降はどんな文字列でも構わないファイル」と解釈され、log log2 の二つのファイルを指します。

任意の一文字に適合するメタキャラクタ「?」

今度は「?」文字を紹介しましょう。例えば上記の例でbbとccの両方のファイルの中身を見たい場合は、以下のようにすれば良いのです。

```
cc2000(92)% more ??
```

つまり??とは「どんな文字でも構わないから、二つからなるファイル」を列挙することを意味しています。その結果more ??はmore bb ccと解釈されて実行されます。

?が適用可能なのは何もファイル名の最後に限りません。例えば上記の例でmore ?bとすればそれは「どんな文字でも構わないから一文字あって、次がbで終わるファイル」と解釈され、bb ファイルを指します。

?文字はファイル名の記述のどこに幾つ現れても構いません。

## ディレクトリに対してのメタキャラクタ

\* と ? 文字はディレクトリに対しても適用可能です。例えば、`ls */*2` などとすれば「カレントディレクトリ以下の全てのディレクトリの下にある、どのような文字列で始まっても構わないから最後は 2 で終わる名前を持つファイル」を見つけてその一覧を表示します。

### 6.4.2 対話的なファイル名の補完

シェルからファイル名を記述しているとき、シェルが提供している対話的なファイル名の補完機能が幾つか利用できます。

再び説明はカレントディレクトリ以下のファイルの状況が以下のようにだと仮定して行います。

```
cc2000(88)% ls
bb      cc      log      log2     sample
cc2000(89)%
```

#### ファイルの名前を途中までタイプしてくれる「<Tab>」

例えば `more sample` とタイプしたくて、`more s` までタイプしたとします。この状況で <Tab> キーを一度押せば `more sample` と、シェルの方で勝手にタイプしてくれて、`sample` から一つ離れた位置にカーソルが移動します。そこで <Return> を押せば良いというわけです。

つまりシェルは `more s` とタイプされた段階で、`s` に続くファイル名としてはこの状況では `sample` しか無いよ、と気を利かせてくれているのです。

今度は `more log2` とタイプしたくて、`more l` までタイプしたとします。この状況で <Tab> キーを一度押せば `more log` と、シェルの方で勝手にタイプしてくれて、`log` に続く位置にカーソルが移動します。ベルが一度鳴るかもしれません。

今度はシェルは `more l` とタイプされた段階で、`l` に続くファイル名としてはこの状況では `log` と `log2` しか無い事が判ります。だからとにかく一致している途中までタイプしてくれているのです。そこで残り不足している `2` をタイプして、<Return> を押せば良いというわけです。

#### ファイル名の一覧を表示する「C-d」

<Tab> による補完を行って、カーソルがファイル名の記述のすぐとなりに来て (ベルが鳴る) 時は、まだ残りがあるかも知れないよ、という意味でした。この段階で、では一体どのような名前のファイルが可能性として残っているのかを確認するには C-d キーを押します。そうすれば以下のように、まだ可能性のあるファイル名の一覧を挙げてくれます。

```
cc2000(86)% more log ... ここで空白などあけずに<Control>-d
log  log2
cc2000(86)% more log
```

こうしてファイル名が長ったらしい場合でも、<Tab> キーと C-d を組み合わせて行けば、少ないタイピングで間違いなくそのファイルを指定することができるというわけです。

### 6.4.3 シェル変数と環境変数

シェルには<sup>7</sup> シェル変数と環境変数と呼ばれる、コマンドなどの動作に影響を与えるものがあります。これを設定し直すことによって、もう少しあなたにとって都合の良い UNIX 環境が得られるかもしれません。ここではシェル変数や環境変数の本質については触れずに、その設定例やヒントを紹介します。本質的なことについては附録の参考文献などからシェルに関する記述を参照してください。

#### シェル変数の表示、設定

現在設定されているシェル変数に何が有るかを調べるには `set` コマンドを利用します。また、シェル変数の内容を変更するにも `set` コマンドを利用します。`set` コマンドには以下の 2 通りの書き方があります。

1. `set`
2. `set var = string`

1. の書き方の場合、現在設定されている全てのシェル変数を表示します。

2. の書き方の場合、シェル変数 `var` の内容を `string` に設定します。

また、`echo` コマンドを使って特定のシェル変数の内容を表示することもできます。`echo $var <Return>` とすると、`var` という名前のシェル変数の内容を表示します。シェル変数は慣例として英小文字と数字で構成されています。

例えば `tcsh` ではしばらく使わないでほうっておいた場合、自動的にそのシェルを終了する機能があります。これを制御しているのはシェル変数 `autologout` です。`echo $autologout<Return>` とする事によって、現在設定されている放置時間が分単位で確認できると思います。これは不用意に切り忘れてしまったような不要なシェルを放置しないための処置なのですが、この時間が短すぎて困る環境の人もいるでしょう。そのような人は、以下のようにして放置時間を延長できます。

```
cc2000(81)% set autologout=300
cc2000(82)%
```

また、例えばカレントディレクトリの情報はシェル変数 `$cwd`<sup>8</sup> に格納されていますので、`echo $cwd` などとして表示させることができます。それからホームディレクトリの情報はシェル変数 `$home` に格納されています。これまた `echo $home` などとして表示させる事ができます。

`echo` コマンドなど、一般のコマンドでシェル変数を扱うときは常に `$` 記号がシェル変数の名前の前に付きます。`set` コマンドでシェル変数の名前を指定するときは `$` 記号を付けないことに注意してください。

#### 環境変数の表示、設定

現在設定されている環境変数に何が有るかを調べるには `env` コマンド<sup>9</sup> を利用します。また、シェル変数の内容を変更するには `setenv` コマンドを利用します。

---

<sup>7</sup> 特に `csch` と、それを模倣している `tcsh` には

<sup>8</sup> `current working directory` の略なのです

<sup>9</sup> UNIX の種類によっては `setenv` コマンドまたは `printenv` コマンドで行うというものもあります。

```
env
```

現在設定されている全ての環境変数を表示します。

```
setenv VAR string
```

環境変数 VAR の内容を string に設定します。

また、echo コマンドを使って特定の環境変数の内容を表示することもできます。echo \$VAR <Return> とすると、VAR という名前の環境変数の内容を表示します。環境変数は慣例として英大文字と数字で構成されています。

例えばプリンタの操作をするときは `lpr -Pcspr01 sample.tex` などして常にプリンタ名を指定するようにはしていました。環境変数 PRINTER が設定されている場合は、プリンタに関する各種コマンドは ( -P オプションを省略した場合 ) 環境変数 PRINTER に指定されたプリンタに対して処理を実行しようとします。

```
cc2000(81)% setenv PRINTER cspr01
cc2000(82)% lpr sample.tex
cc2000(83)%
```

echo コマンドなど、一般のコマンドで環境変数を扱うときは常に \$ 記号が環境変数の名前の前に付けます。setenv コマンドで環境変数の名前を指定するときは \$ 記号を付けないことに注意してください。

#### 6.4.4 リダイレクション

UNIX の標準的なコマンドには、標準入力と標準出力と呼ばれる入出力を処理対象としているものが数多くあります。普通にシェルを使っているときは、標準入力はキーボード入力、標準出力はディスプレイ表示に割り当てられています。

この例として bc コマンドを挙げておきます。bc コマンドを実行して四則演算をキーボードからタイプすると、画面上に計算結果を表示します。これはつまり bc コマンドは標準入力から四則演算を受け付けて、計算結果を標準出力に流しているということの結果です。

さて、標準入力、標準出力はそれぞれ他のファイルに割り当てることができます。この標準入出力のファイルへの割り当て直しを「リダイレクション<sup>10</sup>」と呼んでいます。

##### 標準入力のリダイレクション

標準入力のリダイレクションは「<」記号で表現します。

コマンド [オプション] [引数...] < ファイル名 と書いて、コマンドの標準入力を指定のファイルに割り当て直します。例えば bc コマンドの標準入力をキーボードではなくファイルに割り当ててみましょう。bc < ファイル名 <Return> です。

まず bc.in というファイル名で、四則演算を並べたファイルを用意します。Mule で新規に作成、編集してください。以下にでき上りを cat した例を示します。簡単な四則演算を 3 行用意しました。

```
cc2000(85)% cat bc.in
```

---

<sup>10</sup> redirection 宛名を変える

```
1 + 2
2 * 3
4 / 2
cc2000(86)%
```

bc コマンドの標準入力に対するリダイレクションによってファイルの内容を四則演算の式の列として実行され、画面すなわち標準出力に結果が表示されます。

```
cc2000(86)% bc < bc.in
3
6
2
cc2000(87)%
```

### 標準出力のリダイレクション

標準出力のリダイレクションは「>」記号で表現します。

コマンド [オプション] [引数...] > ファイル名 と書いて、コマンドの標準出力を指定のファイルに割り当て直します。例えば bc コマンドの標準出力をディスプレイではなくファイルに割り当ててみましょう。bc > ファイル名 <Return>です。

```
cc2000(88)% bc > bc.out
1 + 2
2 * 3
quit
cc2000(89)%
```

cat コマンドでリダイレクションによってファイルに結果が残っていることを確認しましょう。

```
cc2000(89)% cat bc.out
3
6
cc2000(90)%
```

### 標準出力のリダイレクションで追加書き

標準出力のリダイレクションを既存のファイルに対して指定すると、単純に上書きしてしまい、そのファイルの元の内容は失われてしまいます。しかし追加書きするようなリダイレクションもあります。標準出力の追加リダイレクションは「>>」記号で表現します。

コマンド [オプション] [引数...] >> ファイル名 と書いて、コマンドの標準出力を指定のファイルに追加するように割り当て直します。例えば bc コマンドの標準出力をファイル追加するように割り当ててみましょう。bc >> ファイル名 <Return>です。

```
cc2000(90)% bc >> bc.out
5 * 2
quit
cc2000(91)%
```

cat コマンドでリダイレクションによってファイルにもとの結果に加えて新しい結果が追加されているのを確認しましょう。

```
cc2000(91)% cat bc.out
3
6
10
cc2000(92)%
```

#### 標準入力、出力の両方のリダイレクション

先ほどの bc.in ファイルを利用して標準入力をそこから、標準出力をこれまで先ほどの bc.out ファイルに割り当てて見ましょう。以下にその例を挙げます。例では cat コマンドでファイルの中身を確認しています。

```
cc2000(92)% bc < bc.in > bc.out
cc2000(93)% cat bc.out
3
6
2
cc2000(94)%
```

### 6.4.5 パイプ

bc などのように、標準入力から何かを受取り、標準出力にその結果を返すようなコマンドをフィルタコマンド<sup>11</sup>と呼んでいます。

先ほどの例で、bc.out というファイルが結果としてでき上がっていると思いますが、例えばこのファイルの中身を数字の小さいもの順に並べ替える<sup>12</sup> 為に、sort コマンドの利用を考えます。sort もフィルタコマンドですから、実行は以下のようにする事になるでしょう。-n オプションはソートの順番を数値の小さいもの順にするためのオプションです。

```
cc2000(94)% bc < bc.in > bc.out
cc2000(95)% sort -n < bc.out
2
3
6
cc2000(96)%
```

単純なコマンドでも、それを組み合わせて実行すると、なかなか便利な<sup>13</sup> ものです。ところで上記のコマンド 2 つの組み合わせを、もっと簡単に表現できるような仕掛けが UNIX には用意されています。以下の例を見てください。

```
cc2000(97)% bc < bc.in | sort -n
```

---

<sup>11</sup> あたかもフィルターのように働くという意味ですね。

<sup>12</sup> 並べ替える作業を一般的に sort (ソート) と呼んでいます。

<sup>13</sup> ちょっと意味のない例しか挙げられませんでした

```
2
3
6
cc2000(98)%
```

2つのコマンドを一行で書いています。リダイレクションを使わないために中間的に発生していた `bc.out` ファイルも必要ありません。このような、コマンドとコマンドをつなぐ「|」記号を「パイプ」と呼んでいます。パイプがあれば、パイプの左のフィルタコマンドは標準出力をパイプの右のコマンドの標準入力につながわせて実行くれます。

例えば `cat` コマンドもフィルタコマンドですから、上の例は以下のようにも書き換えられます。

```
cc2000(98)% cat bc.in | bc | sort -n
2
3
6
cc2000(99)%
```

パイプは幾つでも重ねて使うことができます。

あまり実用的な例ではありありませんが、`echo` コマンドを使えば簡単な計算を標準入力からキーボード入力しないで処理するような組み合わせもできます。

```
cc2000(99)% echo '3 * 4' | bc
12
cc2000(100)%
```

UNIX ではパイプとリダイレクションを応用することによって、各種のコマンドを柔軟に組み合わせて実行することができます。

#### 6.4.6 シェルの鬼へのヒント

以下に少しだけシェルの鬼<sup>14</sup>へのヒントをあげておきます。

##### キーボードからの標準入力の終了

今回用意した `bc.in` ファイルなどは `Mule` を使わなくても以下のようなやり方で作成できます。

```
cc2000(100)% cat > bc.in
```

`cat` コマンドは引数としてファイル名が与えられていない時は、入力を標準入力から行ないますから、それをキーボードから入力して出力先を画面ではなくファイルにするというわけです。

上のコマンドを実行して、タイプする内容は `1 + 2<Return>2 * 3<Return>4 / 2<Return>C-d` です。最後の `C-d` はファイルの終了を意味します。キーボードからファイルの終了<sup>15</sup>を入力するのは、大抵の場合この `C-d` です。

今まで黙っていましたが、`bc` コマンドは、サブコマンド `quit` ばかりでなく、このファイルの終了を受けとっても終了します。だからこそ `cat bc.in | bc` で文句も言わずにちゃんと処理を終了したのです。

<sup>14</sup> ここでキックの鬼と言ってもわかる人は...

<sup>15</sup> 一般的に `end of file` と表現します

## 複数のコマンドを一行に書く

「;」を利用して、複数のコマンドを一行に書くことができます。例えば次のような感じです。

```
cc2000(101)% date ; hostname ; whoami ; id
1994年03月12日(土)16時13分33秒 JST
cc2000
yasuda
uid=2126(yasuda) gid=700(admin) groups=500(clerk)
cc2000(102)%
```

ところで上の4つのコマンドの結果をぜんぶ一つのファイルにリダイレクションしたい場合は、以下のようになります。

```
cc2000(102)% ( date ; hostname ; whoami ; id ) > hostinfo.txt
```

このような書き方で、別々に処理することももちろんできます。

```
cc2000(103)% ( date ; hostname ) > hostinfo1.txt ; ( whoami ; id ) > hostinfo2.txt
```

## 一連のコマンドをファイルに書いておく

シェルもフィルタコマンドです。ですからファイルにコマンド列を書いておいて、それをシェルの標準入力に与えて処理させることができます。cc環境標準のシェルは tcsh ですから、シェルのコマンド名は tcsh です。以下に例を挙げておきます。

```
cc2000(104)% cat batch
echo 'Here is my session information.'
date
hostname
whoami
id
cc2000(82)% tcsh < batch
```

対話形式ではないので警告のメッセージが表示される時がありますが、とりあえず結果は表示されると思います。単純な一連の手続きであればこれで十分ですね。

例のように短いものであれば問題ないのですが、長い複雑なものを作ったりしたときには何らかの覚え書きを入れておきたい時があります。そのような時には # を利用します。# より右に書いたものは覚え書き<sup>16</sup>とみなされて、その部分を一連の実行するべき手続きとして扱わず、無視されます。以下に覚え書きを含めて書いた例を挙げておきます。

```
cc2000(84)% cat batch
#
#   Sample shell program. '95.2 Yasu.
#
echo 'Here is my session information.'
```

---

<sup>16</sup> コメントなどと呼ぶ時もあります。

```
date
hostname
whoami
id      # show my username and etc.
cc2000(85)%
```

#### 6.4.7 シェルよ永遠に

シェルの説明をするというのにあの話（もしくはあの話）をしないとは何事か！と、お怒りの方もおられるでしょう。シェルには非常に多くの機能があります。さすがに全部書くのは無理ですので、これ以降は付録の参考文献を参照してください。

## 第7章 Mule

表記方法として

- C-h            コントロールキーを押しながら h を押す。
- M-x ABC       エスケープキーを押した後に x を押して普通に ABC と打つ。
- M-C-x        エスケープキーを押した後にコントロールキーを押しながら x を押す。
- <Space>      スペースキーを押す。
- <Return>     リターンキーを押す。
- <Delete>     デリートキーを押す。

などというものがあります。それを踏まえて以下を読んでください。

なお、機械の種類によってコントロールキーの機能がオプションキーに変更されている場合があります。(マッキントッシュ等)

### 7.1 なにそれ？

Mule とは、日本語、英語だけでなく、世界中の多くの言語を統一的に同時に扱える様に拡張された Emacs です。Emacs というのは多機能エディタで、単に文章を書くだけでは無く、ネットニュースを読んだり、メールを読み書きしたり、他にも色々な事ができるエディタです。Mule は、その Emacs の機能を全て継承して、更に、多くの外国語を使えるようにしたエディタという訳です。本当にたくさんの機能があり、全てを説明すると本が一冊できてしまいますので、ここでは基本的な Mule の使い方を説明します。

### 7.2 Mule の仕組み

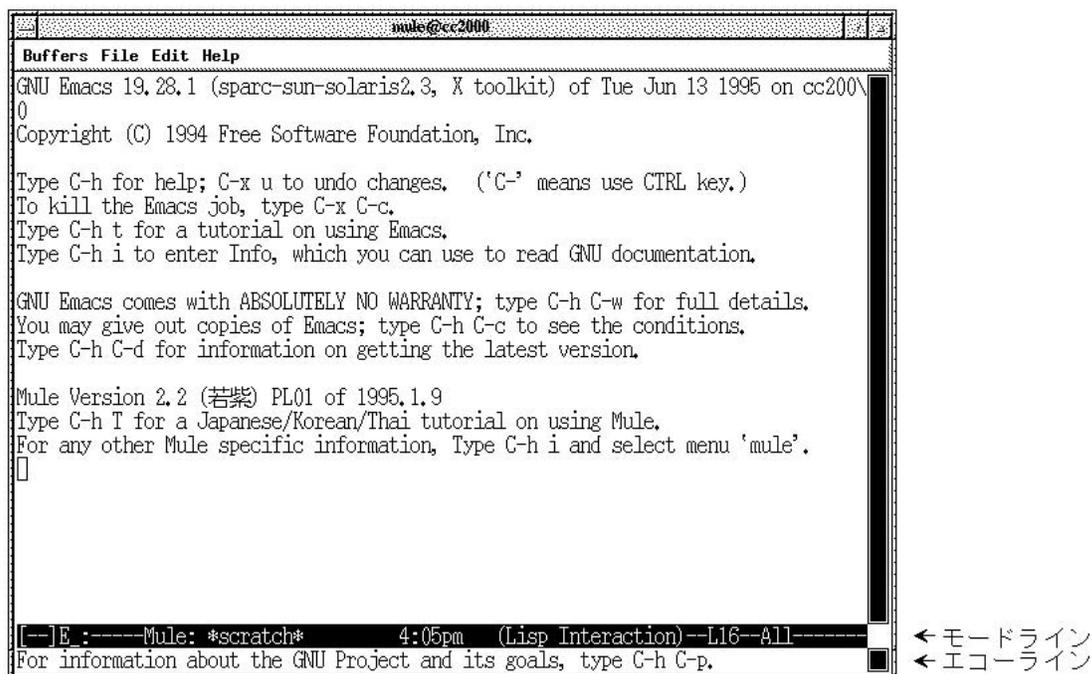
Mule はバッファと呼ばれる作業場所を持ちます。流れとしては、まずディスク上に保存してあるファイルから文書を読み込んでバッファに記憶するか、もしくは新規に何も記憶していないバッファを用意して、それを編集します。そして編集した文章をディスクにファイルとして保存します。

また、Mule はバッファを一つだけでなくたくさん一度に持つことができ、それらを切替えて画面に表示することができます。Mule を起動した時にでてくる画面は「ウインドウ」と呼ばれる大きさが可変のもので、ここにバッファの内容が表示されます。例えば普通に Mule を起動した時はウインドウが一つと、下の方にモードラインという編集中のバッファの情報が表示される行（反転している行）そして一番下にエコーラインという Mule のメッセージを表示する行が表示されます。

このウインドウは幾つにも分割することができ、それらのウインドウに幾つかのバッファを同時に表示させることができます。また、同じバッファを二つ以上のウインドウに表示させることも可能です。その場合、どれかを編集すれば残りのウインドウにもそれが反映されます。

## 7.3 これだけ覚えたら Mule は使える!

これ以降の記述は UNIX の基本的な部分を一通り理解している事を前提として書きます。まず、UNIX のコマンドを入力できるターミナル画面がありますね。では、おもむろに `mule` と打ち込んでリターンキーを押してみてください。えむ、ゆー、える、いー、りたーんです。すると英語ばっかりの画面が現れます。



```
mule@cc2000
Buffers File Edit Help
GNU Emacs 19.28.1 (sparc-sun-solaris2.3, X toolkit) of Tue Jun 13 1995 on cc2000
0
Copyright (C) 1994 Free Software Foundation, Inc.

Type C-h for help; C-x u to undo changes. ('C-' means use CTRL key.)
To kill the Emacs job, type C-x C-c.
Type C-h t for a tutorial on using Emacs.
Type C-h i to enter Info, which you can use to read GNU documentation.

GNU Emacs comes with ABSOLUTELY NO WARRANTY; type C-h C-w for full details.
You may give out copies of Emacs; type C-h C-c to see the conditions.
Type C-h C-d for information on getting the latest version.

Mule Version 2.2 (若紫) PL01 of 1995.1.9
Type C-h T for a Japanese/Korean/Thai tutorial on using Mule.
For any other Mule specific information, Type C-h i and select menu 'mule'.
[]

[--]E :----Mule: *scratch* 4:05pm (Lisp Interaction)--L16-All-----
For information about the GNU Project and its goals, type C-h C-p.
```

図 7.1 Mule 起動画面

ですが別に難しいことが書いてある訳ではありません。この画面は初心者の人にまず知っておいて欲しいことが書いてあります。ヘルプ、言い換えると簡単なオンラインの説明書の出し方やチュートリアルの始め方なんかが書いてあります。

### チュートリアル

チュートリアルというのは日本語にすると「個別指導」で、ここでは画面に従ってキー操作をしながら使い方を覚える入門用の練習のことをいいます。例えば `C-h T Japanese` で日本語のチュートリアルが起動すると書いてあります。チュートリアルをやりたい時はコントロールキーを押しながら `h` を押して、次に大文字の `T` ですからシフトキーを押しながら `T` を押します。そして `Japanese`<sup>1</sup> と入力してリターンキーを押すと始まります。チュートリアルの終わり方はいくつかありますが、Mule ごと終わらせるのが簡単でいいでしょう。`C-x C-c` です。チュートリアルは是非、一度やってみてください。

### カーソルの移動

では本題のエディタの使い方に移ります。超基本部分から覚えていくことにしましょう。カーソルの移動はカーソルキーで行います。矢印の書いてあるキーです。間違えた場合はデリートキーで消します。普

<sup>1</sup> 先頭の J は大文字で後は小文字です。

通はキーボードの上の段の一番右のキーですね。機械によってはバックスペースキーになっている場合もありますのでデリートキーでうまくいかないときは試してみてください。改行はリターンキーです。いろいろとキーボードを触って動作を確認してください。

変になったら

それから、まず覚えておいて欲しいことですが、この先の操作で間違えて、自分の意図しない変なメッセージが表示されてしまった場合は、C-g としてください。これは指示途中の操作を取り消すコマンドです。変になったときは傷の浅いうちにこれをする方が無難です。下手に色々触るとどんどん変になることがよくあります。それでもダメなら周りのよく知っていそうな人に聞いてください。

名前をつけよう

話を戻しましょう。単に Mule を立ち上げた場合 `*scratch*` というバッファが開きます<sup>2</sup>。下の方に `*scratch*` と書かれていますね。これが今編集しているバッファの名前です。

`*scratch*` は「走り書き」の名のとおり落書き扱いですからそのまま `*scratch*` バッファに何か書いても Mule を終了すると保存されずに消えてしまいます。ですからこれを保存するために、まず何か名前をつけてあげないといけません。後から名前をつける方法もありますが<sup>3</sup>、ここでは最初に名前をつけることにします。まず Mule が起動した時に C-x C-f とします。すると

```
Find File: ~/
```

と表示してファイル名を聞いてくるのでそこで何かこのファイルにつける名前を指定します。例えば abc とでもしましょうか。するとエコーラインに (New file) と表示されてモードラインの `*scratch*` と書かれていた場所に abc と表示されますね。

続けて何か書いてみてください。なにか書いてから終わろうと C-x C-c をするとメッセージがでます。

```
Save file /NF/home/g910/ozaki/abc? (y, n, !, ., q, C-r or C-h)
```

これは今のバッファが変更されているのでそれを保存するかどうかを聞いているのです。保存するなら y、しないなら n を押してください。n と答えたときは本当にしないのかどうか確認してくれます。親切ですね。ここでは保存を選びましょう。y、と。他にも選択肢が表示されていますが、今は気にしないでいいでしょう。どうしても気になるのなら C-h を実行して出てくるメッセージを読んでください。

これだけで Mule を使うことはできます。後は知れば知るほど便利に使うことができるのが Mule の特徴で、初心者から上級者まで幅広く自分のレベルにあった使い方ができます。

## 7.4 もうちょっと覚えよう

ファイルの呼びだし

当然、前に保存した続きを書きたい事があります。そういう時は Mule を立ち上げる時に続けて保存したファイル名を打ちます。さっき保存した abc なら mule abc と後ろに続けて abc と打ち込みます。また、ファイルの読み込みにはもう一つやり方があり、Mule が立ち上がった状態で C-x C-f とすると

<sup>2</sup> バッファというのは Mule の作業を行う場所で、一度にいくつも開くことができます。Mule は文書をバッファで編集し、ディスク上のファイルと読み書きします。

<sup>3</sup> C-x C-w で名前をつけて保存することができます。

Find File: ~/

と表示してファイル名を聞いてくるのでそこで指定してやるとそのファイルをバッファに読み込みます。あれ、この C-x C-f というキー操作は先ほど新しいバッファに名前をつける時に行ったものと同じですね。実は C-x C-f は本来ファイルを読み込むための操作なのですが、そのファイルが存在しないと (New file) としてその名前で新しいファイルを作るのです。先ほどの場合は abc というファイルが無かったから (New file) になった訳です。また、この場合のファイル名の入力には補完機能を使う事ができます。補完機能とはファイル名、例えば abc という場合、他に a で始まるファイルがなければ、a だけを入力した時点で <Space><sup>4</sup> を打つと abc とすべて補完されるという便利な機能です。また、他に a で始まるファイルがあればそれをバッファに表示しますので、お目当てのファイルの 2 文字目、3 文字目を入力して、他と区別できるところまで入力したら <Space> でまた補完します。同じ様に M-x *Command* によるコマンドにも補完機能が使えます。

他に、C-x i で前に作ったファイルをカーソル位置に差し込むことができます。C-x C-f と似たようなものですが、読み込んだファイルがカーソルの位置に差し込まれます。

### 大きなカーソル移動

ところで、慣れてきて文章が長くなるとカーソルを移動するのが面倒くさく感じてきます。そういう時は M-v と C-v でそれぞれ 1 ページ分上下に移動します。また、C-a と C-e は行頭と行末への移動です。もっと大きく動きたいなら M-< と M-> でバッファの先頭と末尾へ移動します。これだけ覚えておけば作業が大幅に楽になること請け合いです。

他にも M-x goto-line として行番号を入れるとその行へ飛びますし、今の行番号が知りたければ M-x what-line で表示してくれます。

C-1 も覚えておくと便利です。カーソルを画面の中央に移動するコマンドなのですが画面が乱れた時に書き直す役目も果たしてくれます。UNIX をよく使うようになると、Mule を使っている最中に talk コマンドや write コマンドで誰かから呼びかけられる事もあります。そうすると画面は相手の呼掛けのメッセージでぐちゃぐちゃになってしまい、折角編集していた貴方の文章が消えてしまいます。でも安心してください。これは画面上だけで実際には内容は変わっていないのです。C-1 の画面書き直し機能はこういう時に役に立ちます。後は余り使いませんから必要に応じて覚えるといいでしょう。

### カット&ペースト

次にカット&ペーストを説明しましょう。文字のコピーですね。

まずコピー元を指定します。範囲の先頭で C-<Space> (コントロールを押しながらスペース) でマークをセットします。そして範囲の末尾へカーソルを移動します。このマークからカーソルまでの範囲をリージョンと呼びます。日本語で言うなら矩形 (くけい) ですね。セット位置からカーソル位置までをリージョンとしますので別に末尾でセットして先頭へカーソルをもっていっても構いません。

次に M-w でその範囲の文章を覚えます。また、C-w なら覚えると同時に削除する (カット) ので移動する時や単に消したい時に使えます。場合に応じて使い分けてください。

そしてコピーしたい先までカーソルを移動して C-y で覚えていた文章を挿入 (ペースト) します。マーク位置を忘れたときには C-x C-x でマーク位置とカーソルの場所を入れ替えるのでそれで確認してください。尚、C-x C-x はうまく使うと離れた場所から元の場所へ戻るときに使えます。

---

<sup>4</sup> <Tab>でも補完機能が働きます。<Space>はハイフンなどの区切り記号で一度止まるのに対して、<Tab>は可能な限り全てを補完するのでこちらの方も合わせて使うと便利です。

また、C-k でカーソル位置から行末までをカットします。C-k は行頭で 2 回連続して使うと 1 行消去することができます。これで消去した内容は (連続で消す限り) 覚えていますので C-y で挿入する事ができます。行の移動なんかが楽にできますね。

## Undo

ところで色々編集しているうちに間違えてしまい、元に戻したい時があると思います。そういう時は C-\_ (アンダーバー) でいくらでも元に戻せます。Undo(アンドウ) という機能です。因みに、C-x u でも Undo になります。

## 保存

しかし、いつ何時、どんな理由で編集内容が壊れるか分かりません。一応ある程度書いたら内容を保存しておく習慣をつけておいた方がいいでしょう。精神衛生上もいいですしね。C-x C-s で今のファイル名で保存します。

また、別の名前をつけて保存したいのなら C-x C-w としてください。

```
Write file: ~/
```

このように表示されましたね。ここで好きなファイル名を書けばその名前で保存します<sup>5</sup>。

## 7.5 日本語入力

では次は漢字の入力にいきましょう。mule は Wnn(うんぬ) という日本語変換システムで日本語を入力します。いままでタイピングと言えばアルファベットのタイピングについてのみ説明してきました。しかしやはりひらがなや漢字をファイルの中にかきたいものです。ここでは Mule を用いたひらがなや漢字などのタイピングの方法について説明します。

### 7.5.1 かな漢字変換

アルファベットの場合タイピングは簡単です。つまりキーボード上のキーに書いてある文字が、そのキーを押すことによって入力されるのです。ひらがなのタイピングについては何とかこの方法で済ませられるかもしれませんが、漢字についてはそういうわけには行きません。漢字は数万字 (日常的に使う漢字だけでも数千字) あって、とても数千のキーを並べるわけには行かない<sup>6</sup> からです。

そこで「かな漢字変換」による漢字の入力の登場です。つまりまず「かな」を目標の漢字の読みとして入力することによって、それを漢字に変換しようというアイデアです。最近市販されているワープロは殆ど全てこの方式を採用しています。UNIX コンピュータでもこの方法で漢字をタイプします。しかも「かな」の入力はローマ字からの変換です。つまり「かな」すらキーボードには載っていない (もしくは載っていても使わない) ので、まずアルファベットのキーを利用してローマ字で「かな」を目的の「漢字」の読みとしてタイプし、それを変換するのです。

Wnn<sup>7</sup> は 京都大学数理解析研究所、オムロン株式会社、株式会社アステックの 3 者によって開発されたかな漢字変換システムです。cc 環境ではこの Wnn を標準的な漢字変換システムとして採用しています。

<sup>5</sup> 但し既に存在するファイルを指定した場合、上書きしてしまうので注意してください。

<sup>6</sup> 昔の漢字タイプライタは正にそのとおり数千のキーを並べていましたけどね。コンピュータではその方法は採用していません。

<sup>7</sup> 「うんぬ」と読みます。名前の由来は「私の名前は中野です」の略から来ているそうです。

EGG<sup>8</sup> は電子技術総合研究所の戸村哲氏が中心となって開発した、Mule 上で Wnn を利用するためのシステムです。cc 環境ではこの EGG を Mule 上での標準的なかな漢字変換システムとして採用しています。

## EGG のモード切り替え

さて、Mule 上で漢字をタイプするためには、まず EGG をローマ字かなモードにします。Mule が起動されている状態で、C-\ です。(キーボードによっては C-\ の代わりに、C-¥ かもしれません。)モードラインの左端に注目してください。以下のようになるのが判ると思います。

```
[あ]E_:-----Mule: *scratch*                (Lisp Interaction)--All-----
```

この状態から元に戻るためには、再び C-\ です。モードラインが元に戻りましたね。C-\ でモードラインの左端がくるくる変わるのを確認してください。

```
[--]E_:-----Mule: *scratch*                (Lisp Interaction)--All-----
```

モードラインの左端が [--] となっている状態を EGG の「透過モード」と呼んでいます。(透過モードというのは、つまり今までどうりアルファベットのタイピングがそのまま行える状態です。) [あ] となっている状態を「ローマ字かなモード」と呼んでいます。ローマ字かなモードにしておくと、タイプしたアルファベットはまず EGG に受けとられ、そこでローマ字として解釈されてひらがなが表示されます。それから漢字変換に関するキーを操作して目的の漢字かなまじり文へと変換するのです。

### 7.5.2 ためしに変換

ローマ字かなモードにして、アルファベットで「wata sinonama ehanakanodesu」とタイプしてください。打ち込んだアルファベットが縦棒にはさまれながら次々とひらがなに変換されて行くのが判るでしょう。

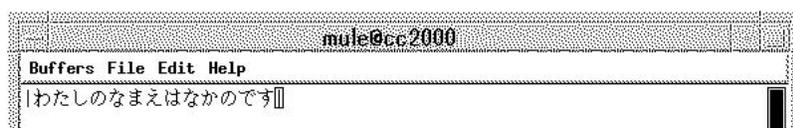


図 7.2 EGG のフェンスモード

この縦棒を EGG ではフェンスと呼び、変換途中のかな文字がフェンスにはさまれている状態をフェンスモードと呼んでいます。

フェンスの中の文字を編集するには普通の Mule の編集のときと同じくカーソルキー( や )と <Delete> キーです。カーソルキーが効かない場合は C-b でカーソル左、C-f でカーソル右と同じ動きをします。

ローマ字が全てかなに変換されたら <Space><sup>9</sup> キーを押すことによって今度はかなを漢字に変換する作業が始まります。

#### 再び注意

あなたがその Mule で初めてかな漢字変換を行うときは、エコーラインに「ホスト local の WNN を起動

<sup>8</sup> 「えっく」と読みます。名前の由来は「たくさんまてさせてごめんなさい」の略「たまご」から来ているそうです。

<sup>9</sup> C-w でも構いません。

しました」などというメッセージが表示されて、かな漢字変換が開始されるのに若干時間が掛かるかもしれません。

あなたが 98 年度以前にユーザ ID の登録をした人なら、初めて使ったときに

頻度ファイル "!\NF/home/syokuin0/ozaki/Wnn6/fisd.h" がありません。作りますか?(y or n) と表示が出るかもしれません。このような場合、y キーを押してください。同様なメッセージが繰り返し表示されますが全て y と答えてください。

かな漢字変換が始まると、モードラインの左端が以下ようになります。

```
[漢]E_:---*-Mule: *scratch*                (Lisp Interaction)--All-----
```

フェンスの中は以下のようになっているでしょう。このモードラインの左端が [漢] となっている状態を「漢字変換モード」と呼んでいます。

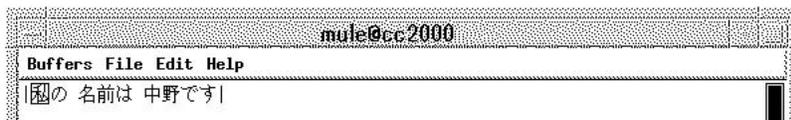


図 7.3 EGG の漢字変換モード

ここで<Return>とすれば現在表示されている漢字かなまじり文の候補「私の 名前は 中野です」が採用されて、Mule にタイプされます。フェンスがなくなって、漢字が Mule の普通の操作で編集できる状態になりましたね。

この作業を「確定」と呼んでいます。これでまず一つ漢字の文章がタイプされました。一安心です。

## 再変換

一度変換するだけで自分の目的の漢字に変換してくれれば良いのですが、先ほどのようにうまく行くことばかりではありません。そこで、EGG が最初に挙げてくれた漢字の候補を再変換しなければなりません。一度でうまく行かない例として、今度は「kyoutonosougakyoukyouwoyomu (きょうとのそうがきょうきょうをよむ)」とタイプして <Space> で変換してください。恐らく以下になるでしょう<sup>10</sup>。

```
|京都の 沿うが 恐々を 読む|
```

ちょっとおかしいですね。ここでは最終的に「京都の僧が今日経を読む」に変換し直す例をあげます。

かな漢字変換では変換は文節単位に行なわれます。はじめはローマ字でタイプされた一連のひらがなの列から EGG が自分で文節の区切りを見つけて変換します。上記の例ではその結果「京都の」「沿うが」「経々を」「読む」と文節の区切りを付けました。EGG の漢字変換モードでは文節の区切りを一つの空白で表します。「京都の」と「沿うが」などの間にそれぞれ一つの空白があるのが判りますね。

## 次の文節に移動

今、カーソルは「京都の」の上にあると思います。ここはこのままでありますから次の文節に移動します。C-f を押してください。「沿うが」の上にカーソルが移動します。もし C-f を 2 度押ししてしまい、行き過ぎてしまったら C-b で戻れます。

<sup>10</sup> 漢字変換を行う度によく使う漢字が先に出てくるようになるので必ずしもこのような変換になるとは限りません。

次の候補を上げる

今、カーソルは「沿うが」の上にあると思います。文節の区切りはこれであっていますから、そこで<Space>を押すことによって次の候補が表示されます。この例では6回<Space>を押すと「僧が」が現れると思います。現れなかったら、現れるまで<Space>を押して下さい。あまり急いで<Space>を押すすぎると肝心の「僧が」が現れても勢いでもう一度<Space>を押してしまう時もあるでしょう。そういう時はC-pで一つ前の候補を再表示する事ができます。

文節の区切りを変える

さて、

|京都の 僧が 恐々を 読む|

こうなりましたね。次の文節に移りましょう。C-fを押して下さい。「恐々を」にカーソルが移ります。さて、ここは文節の区切りがおかしいですね。「きょう」と「きょうを」で区切りたいところです。文節の区切りを縮めるにはC-iです。4回押したら「今日」になりますね。同時に「経を」も自動的に修正されました。これでめでたく望みどおりの結果になりました。<Return>で確定しましょう。

もう一つ文節を伸ばす例をあげます。「ここではきものをぬいでください」と入力して<Space>を押して下さい。

|ここで 履物を 脱いでください|

こうなるといいます。しかし、実は最初の文節が「ここで」ではなく、「ここでは」で区切りたかったのです。そこで文節を伸ばしましょう。C-oで文節を伸ばすことができます。この例では1回C-oを押すと

|ここでは 着物を 脱いでください|

となります。

### 7.5.3 特殊な文字列の入力

ちょっと普通でない文字を入力する必要がある場合の例をあげましょう。例えば、次のように入力したいとします。

Sandai さん：あれチャウチャウチャウ？

xさん：ちゃうちやう。チャウチャウちゃうんちゃう？

次のように入力します。まず、入力モードをアルファベットに変更します。qを押して下さい。モードラインの[あ]が<aa>になりますね。この状態でSandaiと入力します。できたらC-qで元の日本語入力モードに戻します。続いて入力し、san:aretyautyautyau?で「さん：あれちゃうちやうちゃう？」まで入力して下さい。

ここで一度変換しましょう。<Space>を押して下さい。すると

|Sandai 算：あれちゃ 卯 茶宇 茶宇？|

となります。とりあえず「算」をひらがなにしましょう。C-fを1回押してM-hでひらがなに変換できます。続いてC-fを2回押して「あれちゃ」に移動します。文節を短くするためにC-iを2回押します。

|Sandai さん：あれ 茶宇 茶宇 茶宇？|

こうなっていますね。次に C-f を 1 回、C-o を 3 回押して「チャウチャウ」にして、C-f を 1 回、M-h を押して「ちょう」に変換すればできあがりです。<Return>で確定しましょう。

さて、続いては「 x 」の記号ですね。

C-^<sup>11</sup> で記号入力モードにします。最初の一回は少し表示が出るまでに時間がかかります。エコーラインに Garbage collecting...done のような表示がついたり消えたりしますね。点滅が止まったな、と思ったら C-1 を押してください<sup>12</sup>。すると

記号入力: 0. JIS 入力 1. 記号 2. 英数字 3. ひらがな 4. カタカナ 5. ギリシャ文字

エコーラインにこのように現れますので、該当する番号を選んでください。なお、ロシア文字、罫線、部首入力、画数入力、第一水準、第二水準、補助漢字という入力モードもありますのでこれらを使いたい時は C-f や C-b で切替えて選んで下さい。「 x 」は 1. 記号の部類ですから 1 を押して<Return>を押します。C-f ,C-b でカーソルを動かして選択しても構いません。

続いて C-b ,C-f で選べばいいのですが、C-n ,C-p で 1 ページずつ切り替わりますので、C-n を押して次候補を出していくと 7 回ほどで「 x 」が出てくると思います。「 x 」の左に出ている数字、もしくはアルファベットを押して<Return>で確定してください。画面に「 x 」が表示されましたね。もう一度 C-^ を押してください。先ほどの「 x 」の位置にカーソルがありますから同じように C-n を押して、次の候補から「 x 」を追加します。同様に「 x 」を入力してください。

x まで入力できたら「さん：ちょうちょう。」を入力してここまでを<Return>で確定します。

Sandai さん：あれチャウチャウちょう？

x さん：ちょうちょう。

ここまで入力できていますね？後少しです。「ちょうちょうちょうんちょう？」を入力して変換してください。ここまで入力できた人なら簡単に「チャウチャウちょうんちょう？」にできますよね。ちなみに、選択している文節をカタカナにするには M-k とします。

## 補足

C-^ を押すと二度目からは前に入力した記号の位置から始まります。他の種類の記号を入力したい時は C-g で一つ前の選択画面に戻ってください。

また、記号の入力は、例えば「 z1 」なら z1 のように z と何かのキーという形でいくつか登録されています。これは一覧として A.3 で紹介していますので、参考にしてください。

どれでもいいですから一番入力し易いものを覚えてください。

## 7.5.4 まとめ

以下にキー操作のまとめを示しておきます。

C-\ 漢字モードと英語モードの切り替え

<sup>11</sup> うまく動作しないようなら M-x special-symbol-input <Return>としてください。

<sup>12</sup> C-1 は実は画面再表示のコマンドなのですが、最初に記号入力画面にした時は Garbage collecting...done と表示したまま、次の画面を表示してくれないので C-1 を押します。

## ローマ字かなモードでのキー操作

<Delete>	タイプミスの修正、一文字削除
または C-b	カーソル一文字左
または C-f	カーソル一文字右
<Space>または C-w	かな漢字変換の開始

## 漢字変換モードでのキー操作

または C-b	カーソル一文節左
または C-f	カーソル一文節右
C-o	文節を伸ばす
C-i	文節を縮める
<Space>または C-n	次候補の表示
C-p	前候補の表示
M-s	候補の一覧をエコーラインに表示する。数字による選択かカーソルキー（もしくは C-b , C-f , C-p , C-n ）による選択を行い<Return>で確定。 候補一覧表時から抜けて元に戻るには C-g 。
M-k	文節をカタカナに変換する。
M-h	文節をひらがなに変換する。
C-^	記号入力モードにする。数字による選択かカーソルキー（もしくは C-b , C-f , C-p , C-n ）による選択を行い<Return>で確定。 候補一覧表時から抜けて元に戻る時、及び前の選択画面に戻る時には C-g 。

### 7.5.5 ローマ字入力のヒント

「たんい」（もしくは「んあ？」など）のように「ん」の次に母音が来る場合

「ん」を単独でタイプするために N(大文字), n', nn が用意されていますのでいづれかを利用して下さい。「taNi」もしくは「tan'i」, 「tanni」とタイプすれば「たんい」とローマ字変換されます。「たんに」と入力するには「tanmi」と n を 3 回入力する必要があります。

「あっ」（もしくは「んあ？」など）のような小さい「っ」「ぁ」を入力したいのです

小さい「あいうえお」などは、x を前に付けてタイプします。「xa」とタイプすれば「ぁ」とローマ字変換されます。「xtu」で「っ」、「xyu」で「ゅ」です。

#### その他

変換せずにひらがなのまま確定してしまった時はその範囲をリージョン指定して M-x henkan-region、間違った漢字で確定してしまった時は、その範囲をリージョン指定して、M-x gyaku-henkan-region とすると幸せになれることでしょう。

また、変換とは別に通常状態でリージョンを指定して M-x zenkaku-region や M-x hankaku-region でアルファベットや数字を全角、半角に変えられます。

M-x upcase-region <Return><Space>や M-x downcase-region <Return><Space>で大文字、小文

字にすることもできます。

## 7.6 ちょっと中級者

### 7.6.1 単語登録

ちょっと一般的でない専門用語や人名、地名等はなかなか変換できず、漢字1文字ずつ入力しなければならないこともありますね。その単語を1回きりしか使わないのであればそれでもいいですが、何度もそういう単語を使う必要がある場合はちょっと面倒ですね。そういう時は単語を辞書に登録することができます。例えば「神山(こうやま)」に登録してみましょう。まず登録したい単語をリージョン指定します。「神」の位置にカーソルを合わせて C-<Space> でマークをセットして、「山」の次の文字の位置にカーソルを移動させます。ここで、M-x toroku-region としてください。

[あ] 辞書登録『神山』 読み :

エコーラインにこの様に表示されますね。ここで「こうやま」と入力して<Return>を押してください。

登録辞書名: 0.ud 1.muhenkan 2. 3.bunsetsu

するとこのようになりますので、もう一度<Return>を押してください。そうすると品詞名を選択する画面になります。

品詞名: 0. 普通名詞/ 1. 固有名詞/ 2. 動詞/ 3. 特殊な動詞/ 4. 動詞以外の用言/

これらの他に C-n を押せば

品詞名: 0. その他の独立語/ 1. 接頭語, 接尾語/ 2. 単漢字 3. 疑似品詞/

品詞名: 0. その他の特殊品詞/

に切り替わりますので、該当する番号を選択して<Return>を押してください。「神山」は固有名詞ですから最初の1. 固有名詞を選択します。

品詞名: 0./ 1. 人名 2. 地名 3. 人名&地名 4. 固有名詞

するとこのようになりますから、2. 地名を選択して<Return>を押します。これで「登録しました」という旨のメッセージが出て「神山」が辞書に登録されました。「こうやま」と書いて変換してみてください。「神山」に変換されましたね。他の場合、例えば動詞なども同様にして登録してください。

このように普段よく使う単語を登録することでどんどん漢字変換の効率が良くなります。また、それ以外にも Wnn は使用頻度を数えて、高いものを最初に変換するようにしますので使えば使うほど変換効率があがっていきます。最初は変な変換をするかもしれませんが、どんどん辞書を鍛えて使い易いものにしてください。

#### 登録単語の抹消

M-x edit-dict-item としてください。すると

[あ] よみ :

とエコーラインに現れますので消したい単語の読みがなを入力して<Return>を押して下さい。するとずらりとその読みがなで登録されている単語が出てきますので上下移動で消したい単語の行にカーソルを

動かして d を押してください。その行の頭に D というマークがつきますね<sup>13</sup>。消したい単語の行、全てにマークをつけたら x を押してください。すると

いいですか? (yes or no)

と聞いてきますから yes<Return>か no<Return>で答えてください。元の画面に戻って q を押すと辞書編集を終わります。

## 7.6.2 検索

### 検索

検索は C-s とすると文字列を聞いてくるので探したい文字列を入力してください。すると入力した文字列の場所へジャンプします。見つけたい文字列は入力できたけれども、ジャンプした先が期待の物と違う場合は続けて C-s をすると同じ文字列で検索を続けます。行き過ぎた時や上方向に検索したい時は C-r です。目当てのものが見つけられたらカーソルを移動させれば検索モードから抜け出します。もし検索中に元の場所（検索を始めた場所）に戻りたくなったら C-g で戻る事ができます。

尚、検索を終えた後、続けて同じ文字列を検索したい時は C-s C-s、または C-r C-r とするとそれぞれ前の文字列で前後に検索をしてくれるので、再度文字列を入力する必要はありません。

### 置換

置換は M-% の後、検索文字列、置換文字列を入力した後、y で置換、n でパスします。また、!で残り全部を確認せずに置換を行ないます。

## 7.6.3 画面分割

Mule は画面を何分割かして幾つかのウィンドウにバッファを同時に表示することができます。

C-x 2 でカーソルがあるウィンドウを上下に二分割します。それらのウィンドウ間のカーソル移動は C-x o です。また、C-^ でカーソルのあるウィンドウサイズを縦方向に拡大します。逆にバッファ表示を減らすときは C-x 1 でカーソルのあるウィンドウ以外を隠しますし、C-x 0 (ゼロ) でカーソルのあるウィンドウを隠します。ただし、隠すだけで完全に消去はしませんし、保存もしません。注意してください。

ふたつ以上のウィンドウに同じバッファを読み込むことも可能ですし、違うバッファを C-x C-f で読み込んで同時に編集することも可能です。前者の場合はそれぞれのウィンドウのバッファは違う場所を表示できますが実体は同じなのでどちらかを編集すると他のウィンドウの同じバッファにもそれが反映されます。

## 7.6.4 バッファリスト

Mule は同時に幾つものバッファを扱えます。C-x C-b で現在のバッファリストを表示させることができ、バッファを切り替えるには C-x b でバッファ名を続けて指定します。幾つものバッファを開いている時はバックグラウンドのバッファの状態（保存したか否か）に注意してください。画面分割と併用すると違う文章からのカット&ペーストが簡単にできます。

---

<sup>13</sup> ただし削除できるのは private 辞書に登録したのだけです。

### 7.6.5 モードライン

モードラインとは Mule の下の方にある \*scratch\* 等のバッファ名やそのバッファの状態を表示する行です。例えば普通に Mule を立ち上げたときは

```
[--]E_:-----Mule: *scratch*          (Lisp Interaction)--All-----
```

こう表示されます。これは前から順に説明すると、

[--]

日本語入力の際に使われる領域です。

E\_

バッファのファイルの読み書きに使用する漢字コードを表します。色々ありますが、ここでの E\_ は \*euc-japan\* を表しています。EUC は Extended Unix Code の略で、主として UNIX の世界で広まってきたコードで、cc 環境の Mule では何も指定しなければこれが用いられます。JIS は通信の方でよく用いられ、J\_ と表されます。これはいわゆる日本工業規格ですね。中には絶対 JIS コードで書かなければいけないファイルなどもあります。何かパソコンとデータをやりとりする人は SJIS も使うことがあるかもしれません。これは Shift JIS の略で S\_ と表されます。DOS と PC の世界で使われています。

ファイルへの書き込みの際の漢字コードを変更するには C-x C-k f です。<Space>を押すと一覧が現れますので目的のコードを選んでください。

-----

バッファの変更状態を表す領域。変更されてまだ保存されていない時は ---\*-、書き込み不可の時は --%- となります。

\*scratch\*

バッファ名です。

Lisp Interaction

この位置には Mule の動作モードを表示します。Fundamental モードや C モード、Fortran モード、TeX モード、gnus モード、mh-e モード、dired モード等、Emacs-Lisp と呼ばれる言語で用意された、たくさんのモードがあります。

All

All はバッファの全てが画面に表示されていることを示し、他に Top(バッファの始めが表示されている)、Bot(最後が表示されている)、num% (全体を 100%としてどのあたりが表示されているか) があります。

### 7.6.6 Dired モード

Mule のファイラー (ファイル操作) 環境です。ファイルの呼びだしやコピー、リネームや消去などが簡単にできます。

M-x dired <Return> で Dired モード に移ります。(ファイル選択時にも起動します。)

カーソルキーで移動して目的のファイルまで移動します。ディレクトリを移動したい時はディレクトリにカーソルを合わせて f を押してください。ちなみに終了は C-x k です。

以下が目的のファイルの位置でできる機能一覧です。

- f カーソルのあるファイルを読み込む
- c カーソルのあるファイルを別ファイルにコピーする
- r カーソルのあるファイルを別の名前に変更する
- d ファイルに消去する印を付ける
- # # 印の付いた自動セーブファイル全てに消去印を付ける
- ~ ~ 印の付いたバックアップファイル全てに消去印を付ける
- u ファイルに付いた消去印を消去する
- x 消去印の付いたファイルを実際に消去する

便利な使い方としては `honyarara~` のようなバックアップファイル<sup>14</sup> をまとめて消したい時です。そのディレクトリで `Dired` モードの `Mule` を立ちあげて、`~` を押して、続けて `x` を押せば本当に消すかどうか、消すファイルの候補リストと共に確認をとってくるので、`yes` と答えればそれらを消してくれます。手で `rm` コマンド等を使うより、速く、間違いなく消すことができます。

## 7.7 上級者に向けて

`Mule` は個人個人が自分に使いやすいようにカスタマイズする事ができます。ただしカスタマイズに不具合があると最悪、`Mule` が動かなくなったりします。変更するときは十分な知識を持って自分の責任で行ってください。ここでは詳しく触れずに幾つか簡単な例を A.3 で挙げておきます。後は必要に応じて勉強してください。

## 7.8 X 環境での Mule

X 環境での説明に入る前にマウスについて触れておきます。`Mule` はボタンが 3 つ付いたマウスで使えるように設計されています。その 3 つのボタンを、以後左ボタン、中ボタン、右ボタンと呼びます。さて、ボタンが 3 つ付いたマウスを使っている場合はそれで良いのですが、2 つ、ないしは 1 つのマウスもあります。ボタンが 1 つしかないマウスの場合は以下の説明の内、左ボタンの機能しか使うことができません。しかしボタンが 2 つの場合は両方のボタンを同時に押すことで、中ボタンの代わりになります。

X 環境で `Mule` を使う場合、そうでない環境に比べて、少し便利に使うことができます。

一番使い勝手が違うのは文字カーソルの移動がマウスでできることでしょうか。移動先にマウスカーソルを合わせて左クリックでそこに文字カーソルを移動させることができます。また、ボタンの役目は以下のとおりです。(実はこれは X 環境の機能です。)

- 左ボタンでのドラッグ      その範囲をリージョンとして記憶します。(M-w)
- 中ボタン                      コピーします。(C-y)
- 右ボタンのダブルクリック      文字カーソルからマウスカーソルまでをリージョンとしてカット。(C-w)

次に、X 環境の `Mule` は、ウインドウの右側にスクロールバーがつきます。ここにマウスカーソルを合わせて左クリックで下方向にスクロール (C-v)、右クリックで上方向にスクロール (M-v)、中ボタンでバーを掴んで上下スクロールができます。

<sup>14</sup> `Mule` はファイルを上書きする時に、前の古い内容のファイルを後ろに をつけたファイル名で置いておきます。これをバックアップファイルといいます。

もう一つ。ウインドウの上側にメニューがついていますね。Buffers,File,Edit,Help の4つです。ここにマウスカーソルを合わせて左クリックしてみるといろいろな項目が現れます。Open File(C-x C-f ) 等、普段よく使うキー操作が登録されています。これを利用すればキー操作を覚える必要がなくなります。ワンタッチで日本語チュートリアル(練習)やネットニュースも立ち上がるので便利です。

他にも色々なモードの時の選択画面などの時にマウスで選択できたりします。

## 7.9 とりあえずここまで

Mule は言い出すときりがありません。実際、Mule の解説書は何百ページもあります。しかし実際使う機能はここまでで取り上げた機能まで十分という方がほとんどだと思いますので、ここでの解説はこれぐらいで終わらせていただきます。A.3 に代表的な機能の一覧を載せておきましたので参照してください。もっと Mule を使い込んでみたいという方は一般に本が発売されています。参考文献にも何冊か紹介されています。図書館にもいろいろ本がありますのでそちらも参照すると良いでしょう。

# 第8章 はじめてのHTML: WWWページを作ろう

## 8.1 はじめに

Netscape Navigator (以下「Netscape」) のような WWW ページを見るためのブラウザ (閲覧ソフト) であちこちの WWW ページを巡っていると (つまり「ネットサーフィン」していると)、「おーっ!」と思うような良くてきたページに出会うことがありますよね。そういうページを見ると、自分でも作りたくなってきませんか? 京都産業大学の cc 環境では、WWW ページは思ったよりも簡単に作ることができます。ここでは、WWW ページの基本的な作り方を記します。あなたも、WWW ページを見ているだけでなく、魅力的な情報を世界に発信しましょう!

さて、WWW ページは思ったよりも簡単にできると書きましたが、WWW ページを書くための言語があるので、それに従って書く必要があります。いえ、言語と言ってもそれほど難しくはありません。WWW ページを書くための言語のことを HTML と言います。HTML とは、HyperText Markup Language の略です。なんだか難しそうですが、分かりやすく言えばこういうことです。WWW でネットサーフィンしていると<sup>1</sup>、あるページからあるページにジャンプしているように見えますが(「リンク」していると言います)、そのような仕組みになっているテキストをハイパーテキスト (hypertext) と言います。そして、ハイパーテキストを可能にするように文書の中で指定することをマークアップ (markup) すると言います<sup>2</sup>。そのようにハイパーテキストをマークアップする言語のことを HTML と言います。

## 8.2 HTML を書く前に : 忘れちゃいけないこと

いよいよ WWW ページを書くことになります。しかしながら、自分の書きたいこと、自分の入れたいものを何でもかんでも入れて良いというわけではありません。一つ気を付けておかなければならないことがあります。それは、著作権法などの法律を侵さないようにするということです。「著作権って何?」という人もいますので、WWW ページに入れても良いものと入れてはいけないものを少し具体的に書いておきます<sup>3</sup>。ただし、これは一例ですから、分からない場合にはネットニュースで聞いてみたり、法律に詳しい人に尋ねると良いでしょう。

### 8.2.1 WWW ページに載せても良いもの

WWW ページに載せても良いものとしては、例えば次のものが考えられるでしょう。

- 自分で書いたエッセイやレポートや論文
- 映画やコンサートや CD などの感想

---

<sup>1</sup> WWW (= World Wide Web) については、「WWW」の章をご覧ください。

<sup>2</sup> L<sup>A</sup>T<sub>E</sub>X も本文以外に色々な命令を埋め込まなければいけないので、マークアップする言語の一種ですね。

<sup>3</sup> 著作権法は、巻末にも掲載されていますので、ご覧ください。

- 自分で撮った写真、自分で撮った映画、自分で描いた絵
- 自分で作った音
- 自分で作ったプログラム

など、要するに自分で作ったオリジナルなものはどんどん入れましょう。ただし、自分が撮ったものでも写真や映画などは肖像権の問題もありますから、自分以外の方が写っている場合には、その人に WWW ページに掲載しても良いかどうかを聞いて許可を得た上で載せましょう。そうしないと、訴えられることもあるかもしれませんよ。

### 8.2.2 WWW ページに載せてはいけないもの

WWW ページに載せてはいけないものとしては、例えば次のものが考えられます。

- 雑誌や本のグラビアなどから取った写真や絵
- アイドル歌手などの CD などからサンプリングした音
- 他人の曲の歌詞
- 他人が書いたコラム、新聞記事や WWW ページ

など、要するに他人が作ったものを無断で自分の WWW ページに入れてはいけません！これらは著作権法違反という立派な法律違反で、当然罰則もあります。

残念ながら、色々な WWW ページを見てみると、この著作権法が守られていないページもあるようです。このようなページを作ることは、許されません！京都産業大学の中でその様なページが見つかった場合には、ページのオーナーであるあなたに必ず謝罪と削除などのしかるべき対応を求めます<sup>4</sup>。結局、他人が作ったものの権利を守ることは自分が作ったものの権利を守ってもらうことにもなりますし、逆のことも言えることを覚えておきましょう。

## 8.3 HTML を書くための準備

### 8.3.1 WWW ページ用のディレクトリはどこ？

このことに関しては、「WWW」の章や、京都産業大学の WWW ページからたどれる「各種ドキュメント」の中の「WWW 個人ページを作ろう」に詳しく書いてありますので、そちらを見てください。URI<sup>5</sup> は、

`http://www.kyoto-su.ac.jp/information/Howtomake_Yourpage.html`

です。しかし、少しだけ説明しておきます。

あなたの WWW ページ用のディレクトリはどこにありますか？何ですって？まだない？それでは、作らなければいけないですね。cc 環境の UNIX コンピュータに login した後に、`wwwmkdir` コマンドを実行してください (1998 年 3 月時点)。

<sup>4</sup> 残念ながらこれまでもそういうことがありました。

<sup>5</sup> Uniform Resource Indicator の略です (以前は Uniform Resource Locator と言われていましたが、HTML4.0 という最新の規格でこのような名称に変更されました)。とりあえず、WWW ページなどの住所だと思っておいたらよいでしょう。

```
cc2000(83)% wwwmkdir
```

これを実行すると、WWW ページ作成に関する注意書きが出てきますので、しっかりと読んでください。そこには、著作権法を侵さないようにということが書いてあります。それに同意できれば、y を押してリターンを押します。成功したならば、しばらく待つとどこのディレクトリにどういう名前のファイルができたかをお知らせしてくれます。できるファイルの名前は、index-j.html です。このファイルは、例えば外国語学部の言語学科のユーザ IDatake 君の場合には次のディレクトリにできたと教えてくれます (ユーザ ID の atake の部分は、あなたのユーザ ID に置き換えてくださいね)。

```
/NF/home/g970/atake/public_html/index-j.html
```

詳しいことについては、先ほど紹介した `Howtomake_Yourpage.html` をご覧ください。

`wwwmkdir` で作成したページは、`wwentry` というコマンドを実行しないと「京都産業大学の WWW ページ」の中の「個人の WWW」の中の各学部学科別氏名一覧の中に登録されません。詳しいことについては、先ほど紹介した `Howtomake_Yourpage.html` をご覧ください。

さて、`wwwmkdir` というコマンドを使ってできたあなたのページを一刻も早く見たいことでしょう。ブラウザで見たい場合には、「Open」ボタンを押して、例えばですが次のような URI を入れてください。もしも Open ボタンがなければ、画面の上の方にある「Location」のところに打ち込めば OK です。

```
http://www.kyoto-su.ac.jp/~atake/index-j.html
```

どうですか？ あなたの名前が入ったページがきちんと表示されましたか？これで、あなたも WWW ページのオーナーです！ :-)

### 8.3.2 Mule は使える？

さて、`wwwmkdir` が作ってくれた `index-j.html` というファイルの中身を見てみましょう。

まずは、そのディレクトリまで `cd` コマンドで移動しなくてはなりません。先ほどの言語学科の atake 君の場合には、現時点では次のコマンドで移動します

```
cc2000(84)% cd ~/public_html/
```

そして、それを Mule で読み込みましょう。読み込み方は「Mule」の章を参照してください。

これで第 1 の準備は整いました。もっとも、この `index-j.html` ファイルも見ているだけでは足りなくなって、書き加えたりしたくなるかもしれませんね。書き加えていきましょう。しかし、後の 8.4 で述べるように、具体的に WWW ページを書いていく段階になると、ある程度 Mule が使えないと書くのにもものすごく時間がかかるかもしれません。ですので、「Mule」の章を見て、Mule を使いこなせるようにしておくと、ページ作りが楽になります<sup>6</sup>。

### 8.3.3 Netscape などのブラウザは使える？

第 2 の準備は、Netscape などのブラウザを立ち上げて、先ほどできた `index-j.html` を読み込むことです。そうしないと、本当に修正されているかどうか分かりません。肝心のブラウザを使うことができま

---

<sup>6</sup> 必ずしも cc 環境の UNIX コンピュータの Mule を使わなければならないということはありません。自分で Mac などを持っている人は、エディタ (Mac なら例えば Nitemacs、Jedit など) で作ってテキストファイルにさえしておけばそれを転送することで使い回しが効きます。しかし、Mac から cc 環境の UNIX コンピュータに送ったり色々に変換したりする手間が必要ですので、京都産業大学では最初から cc 環境の UNIX コンピュータの Mule で書く方が楽でしょう。

すか? Netscape であれば、UNIX 版でも Mac 版でも Windows 版でもどれでも良いので、立ち上げておきましょう。勿論、Microsoft Internet Explorer (以下「MSIE」) でも構いません。

立ち上げ方は 160 ページ、5.6.4 を見てください。

### 8.3.4 忘れずに! Mule で Save した後にブラウザで Reload

これで、準備は整いました。Mule で index-j.html を読み込んで、ブラウザで WWW ページを読み込んでいますね? 後は、この後に記す方法を色々使って、自分のプロフィールなどを書き足していけばよいわけです。

そして、Mule で書き込んだり修正したら、必ず C-x C-s で Save (セーブ、「保存」ですね) して、その後でブラウザの「Reload」(「再読み込み」のことです) ボタンを押して、もう一度読み込み直します。そうすると、Mule で変更した部分がきちんと変更されて表示されるはずですが、これを繰り返すことになり、人によりますが、満足のいく出来上がりになるまで、結構何度もこの作業を繰り返すことになると思います。

ところが、こういうこともあります。Mule でファイルを変更したにもかかわらず Save するのを忘れて、ブラウザだけ一所懸命 Reload して、「あれ? 変わってへん。何でや? 」とつぶやいている人を見かけることが時々あります。しかし、Mule で Save しない限り、何度 Reload をかけてもいつまでたっても変わらないままでしょう。;-) ですから、「Reload の前に、まず Save」を忘れずに、WWW ページを書いていきましょう。

## 8.4 これだけで WWW ページは書ける: 基本編

これから、WWW ページの基本的な書き方を記していきます。京都産業大学の共用のコンピュータではほぼ全てで Netscape が使えますので、Netscape を中心にしながらその他のブラウザで表示してもおかしくないような HTML の書き方を記します。ただ、WWW の世界は刻々と変化しており、HTML の規格も変わりつつあります。あまりややこしいことを言うつもりはありませんが、昨年の終わりに発表された最新の規格をも考慮して記しています<sup>7</sup>。より標準的とされているページを書きたい人は、参考にすると良いでしょう。

なお、表示のサンプルには、Mac 版の Netscape Navigator 英語版 2.02 の画面を使っています<sup>8</sup>。Netscape Navigator といってもバージョンによって表示が異なる場合がありますし、同じバージョンでも UNIX 版や Windows 版などの画面とは違う場合があることを、あらかじめお断りしておきます。

### 8.4.1 ファイル名の付け方

先ほど、`wwwmkdir` を実行した時に、`index-j.html` という名前のファイルができました。この最後に付いている `.html` というのは、おまけで付いているわけではありません。これは拡張子と言い、そのファイルがどういう種類のファイルであることを示しています。`html` 自体の意味はもうお分かりですね? そして、この `.html` が付いていることによって、ブラウザはそのファイルが HTML のファイルであることが認識できるわけです<sup>9</sup>。ですので、新しいファイルを作りたい場合にも、例えば `newfile.html` のように必

<sup>7</sup> 1997 年 12 月 18 日に発表された「HTML4.0 W3C 推奨」です。URI は、<http://www.w3.org/WWW/TR/> です。もちろん英語で書かれています。;-p 個人的に作られた日本語バージョンもあります

<sup>8</sup> 最新バージョンは 4.04 ですが、文章の論理構造を見るためだけにこれで十分と言えるでしょう。

<sup>9</sup> また、`.html` という拡張子を持ったファイルを Mule で読み込むと、8.6.1 に記す `html-helper-mode` に自動的に入るので、それを使ってページを楽に書くことができるようになります。

ず.html を付けてください<sup>10</sup>。

## 8.4.2 「タグ」って何？

index-j.html ファイルを Mule で読み込んで見てみると、<HEAD>のように<と>で囲まれたものがたくさん出てきます。このようなものを、いわゆるタグ (名札) と言います。このタグは、ブラウザで見ると表示されていません。表示されませんが、その代わりに「字を大きくせよ」「改行せよ」などの命令や様々な設定を行なう、いわば「黒子」の役割をしています。しかし、もしもこのタグが全くなければ、全部の文章が一続きに表示されてしまい、とても見にくくなるでしょう。そういう意味で、非常に重要な裏方さんなのです。そして、このタグをいかにうまく使いこなすかによって、あなたのページが見やすくも見にくくもなります。

このタグには、2種類あります。1) ペアで使うタグと、2) 単独で使うタグです。ペアで使うタグとは、

```
<B>太字になる</B>
```

のように文字列を囲むタグです。ペアで使うタグは、<atake>で始まったら、</atake>のように斜め線 (スラッシュ) 付きのタグで終らなければなりません。そうしないで、</atake>を書き忘れてしまうと、最初のタグに含まれている命令がずっと続いて、変なことになってしまいますので、忘れないようにしましょう<sup>11</sup>。一方、単独で使うタグ (「空タグ」とも言います) とは、

```
<BR>
```

のようなタグです。そして、非常に大事なことは、

タグは半角 (つまり英数モード) で書かなければいけない！

ということです<sup>12</sup>。本文を日本語で (つまり全角で) 書いている時にタグを入れようとして、そのまま日本語モードで入れると、< HR >のようなおかしいタグになってしまい (どこがおかしいか分かりますか?)、せっかく書いたつもりでもブラウザは言うことを聞いてくれません。本来裏方さんなのに、全部表示されてしまいます。必ず半角 (英数モード) で入力しましょう！<sup>13</sup>

## 8.4.3 重要なタグ

さて、index-j.html ファイルを Mule で見てみると、ブラウザでは見えなかったタグが色々出てくることが分かります<sup>14</sup>。その中でも、以下に記してあるタグは重要なタグです。ですから、消したりしないようにしましょう。まずは見てください。その後で、一つ一つ見ていきましょう。

<sup>10</sup> なお、MS-DOS のように拡張子が 3 文字に制限されているような場合には、.htm しか付けられませんが、それでもきちんと認識されます。しかし、cc 環境で HTML のファイルを作っているならば、.html を付けておきましょう。

<sup>11</sup> Mule では、8.6.1 に述べる html-helper-mode を使うと、タグの閉じ忘れをかなり防ぐことができます。:-)

<sup>12</sup> ただし、タグの中でも”と”の間には、日本語を入れることはあり得ます。

<sup>13</sup> 8.6.1 に記す html-helper-mode を使うと、日本語モードに入っている、ある特定のキー操作だけでタグが半角で入るので非常に便利です。:-)

<sup>14</sup> ブラウザでも、「View」メニューの中の「Document Source」を使えば別です。これなら、Mule で読み込んだ時と同じように見ることができます。ただし、編集はできないので「見るだけ状態」ですけれども。

## 入力

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0//EN">
<HTML>
<HEAD>
<TITLE> This is my cool homepage :-)</TITLE>
</HEAD>
<BODY>
ここから本文。書きたいことはここに書く。
</BODY>
</HTML>
```

DOCTYPEって？

いきなり何だか難しそうですが、この一行を DOCTYPE 宣言と言います<sup>15</sup>。ともかく、この一行は必要だということだけ覚えておいてください。最初の方で記した HTML の最新の規格ではこの DOCTYPE 宣言は必須 とされ、今後はブラウザがこの一行を見て「お、ほな、こんなふうに表示したろ」という状況になる可能性もあるので、消さないようにしましょう<sup>16</sup>。

<HTML>と</HTML>

<HTML>と</HTML>は、ファイルの最初と最後に置きます。これは、このファイルが HTML ファイルであることを示すものです。

<HEAD>と</HEAD>

<HEAD>と</HEAD>で囲まれた部分は、本文には現れませんが、次に書くタイトルを表すタグを入れます。

<TITLE>と</TITLE>

<TITLE>と</TITLE>で囲まれた部分には、そのファイルのタイトルを書きます。ここで書かれた内容は、ウィンドウの上の方にタイトルとして現れたり、ブックマーク (しおり) で登録するときの名前になったり、ブラウザでそのファイルを保存するときのタイトルになります。重要な部分ですので、分かりやすいタイトルを付けましょう。また、ブラウザによっては日本語に対応していなくて「文字化け」する場合がありますので、現時点では英数字を使う方が無難でしょう。さらに、先程述べた HTML の最新の規格では、この TITLE タグも必須 とされましたので、必ず書いておきましょう。

<BODY>と</BODY>

<BODY>と</BODY>の間には、本文を書きます。このタグの間以外しか、本文はとりあえず書けないとおいた方が良いでしょう。

<sup>15</sup> この一行はもしかしたら index-j.html のファイルにはないかもしれませんが、ただ、新たに.html がついたファイルを作成するとこの一行が先頭にあると思います。

<sup>16</sup> もし書かれていない場合には、HTML にある程度慣れた段階で自分で追加しても良いでしょう。

上に書いたタグは重要なものですから、cc 環境の UNIX コンピュータで新たにページを作る場合、例えば hoge hoge.html というように.html がついたファイルを Mule で作ると、自動的に入ってくれます。cc 環境の UNIX コンピュータ以外で新たにページを作る場合には、上の重要なタグをあらかじめ書いたファイルを sample.html のように作っておいて、それをひな形にしておいたりコピーして名前を変えて中身を書き換えるようにすると良いでしょう。

このファイルをブラウザで見ると、次のようになっています。上のファイルの中身のどの部分がどのように表示されているかを、図 8.1 で確認してみましょう。



図 8.1 絶対なければいけないタグたちは、表示されない

<ADDRESS>と</ADDRESS>

その他に、そのページを作ったのが誰かということを入れておくタグも書いておいた方が良いでしょう。<ADDRESS>と</ADDRESS>の間に、次のように自分の名前や電子メールのアドレスを入れておきます。これは、図 8.2 のように自動的に斜体の文字になります。また、大抵は</BODY>の直前のあたりに書きます。

入 力

```
<ADDRESS>竹内茂夫 (atake@cc.kyoto-su.ac.jp)</ADDRESS>
```

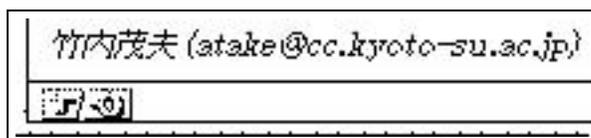


図 8.2 アドレスの表示

#### 8.4.4 特殊文字：そのままでは出ない文字

HTML ファイルの中身(「ソース」とも言います)を見ると、タグが出てきますが、それらは<と>に囲まれています。しかし、<と>で囲まれたものは、命令として解釈されるので、ブラウザで見ても表示されません。

ですので、<と>などを表示させるためには、特別な書き方が必要です。このような特別な扱いをするいわゆる特殊文字 (予約文字) には、<と>の他に、&と"があります。これらの特殊文字は、次のように書くことによって文字としてブラウザに表示されます<sup>17</sup>。

表示したい文字	:	書き方
< (小なり)	:	&lt;
> (大なり)	:	&gt;
& (アンパサンド)	:	&amp;
" (二重引用符)	:	&quot;

#### 8.4.5 コメント (書くけれども表示させない)

WWW ページを書いている、一旦書いたけれどもやっぱりやめたい...でも、せっかく書いたんだから残しておきたいと思うこともあるでしょう。そういう時には、<!--と-->で囲むことによって、ソースには書いているけれどもブラウザでは表示させないようにすること (コメントアウト) ができます<sup>18</sup>。

入力

```
<!-- ここはコメントで表示されないから、何を書こうかな。;-) -->
```

#### 8.4.6 改行する

HTML も L<sup>A</sup>T<sub>E</sub>X の場合と同様に、改行は命令を書くことによってはっきりと示してやらなければいけません。いくら元のファイルの方で Return キーを何回も押して改行しても、ブラウザで見ると全く改行されていません<sup>19</sup>。改行は、<BR>で示します。

入力

```
ちょっと改行したいな。  
よっころしょっと。あれ？ これだと改行できない。:-(<BR>  
ちょっと改行したいな。<BR>  
よっころしょっと。これで、改行できた。:-)
```

↓

#### 8.4.7 段落を区切る

ブラウザで WWW ページを見ていると、段落は紙に書く場合のように次の段落の頭を一字下げのよりは、一行分ほどの空白で区切られていると見やすいようです。このためには、改行して少し空けたいところに<P>と書きます。

<sup>17</sup> しかしながら、ブラウザによっては文字にならずに書いたとおりに表示されます。例えば、&を表示するために&amp;と書いても、&と表示されずにそのまま&amp;と表示されるという悲しい事態もあり得ます。

<sup>18</sup> ただし、L<sup>A</sup>T<sub>E</sub>X の%などのコメントアウトする記号とは違って、万能ではありません。せっかく<!--でコメントアウトを始めても、その後に>があればそこでコメントが終ってしまい、その後は表示されてしまいます。その後もコメントアウトしたければ、再度<!--と-->で囲んでやらなければいけません。これは何とかして欲しいところです。:-)

<sup>19</sup> L<sup>A</sup>T<sub>E</sub>X ならば 1 行空ければ改行されますよね。

ちょっと改行したいな。よっこらしょっと。あれ？これだと改行できない。:-(  
ちょっと改行したいな。  
よっこらしょっと。これで、改行できた。:-)

図 8.3 改行する

入 力  
これで一行分空く。  
<P>さあどうだ！



これで一行分空く。  
さあどうだ！

図 8.4 段落を変える

#### 8.4.8 区切り線を入れる

段落と段落を分ける場合に、<P>を使えば少し空きますが、別の方法として、水平の区切り線を引くこともできます。そのためには、<HR>と書きます。

入 力  
区切り線が引ける。  
<HR>  
なるほどなるほど。



区切り線が引ける。  
\_\_\_\_\_  
なるほどなるほど。

図 8.5 区切り線を入れる

#### 8.4.9 見出しを付ける<sup>20</sup>

文章を書いたら、標題などを大きくたい文字で何か見出しをつけたいですね。また、段落に分けたときも、ちょっと大きな文字で段落の見出しをつけたいこともあると思います。逆に、ちょっと小さな見出しをつけたいこともあるでしょう。そのためには、見出しの大きさを変えたい文字を<Hn>と</Hn>で囲みます。<Hn>のnには1から6までの数字が入ります。1がもっとも大きく、6がもっとも小さい見出しになります。

このタグは見出しのためのタグですので、文字の大きさと種類が一緒に変わってしまいます。また、このタグを使うと、前後の行に必ず空白が空きますので、1行の中のある文字だけ大きさを変えるということではできません。

入 力

```
<H1>big な見出しから small な見出しまで</H1>  
<H2>big な見出しから small な見出しまで</H2>  
<H3>big な見出しから small な見出しまで</H3>  
<H4>big な見出しから small な見出しまで</H4>  
<H5>big な見出しから small な見出しまで</H5>  
<H6>big な見出しから small な見出しまで</H6>
```

↓

big な見出しから small な見出しまで  
big な見出しから small な見出しまで

図 8.6 見出し

#### 8.4.10 文字を修飾する<sup>21</sup>

<B>と</B>

文章を書いていると、文字を大きくしたりする代わりに、これは太くしたい、ということもあるのでないかと思います。HTML では、太くするには<B>と </B>で囲みます。

<sup>20</sup> ただし、cc 環境の UNIX コンピュータの Netscape の日本語フォントでは正確に対応していない場合があります。

<sup>21</sup> ただし、cc 環境の UNIX コンピュータの Netscape の日本語フォントでは対応していない場合があります。

<I>と</I>

欧文の文章を読んでいると、強調部分や書名などが斜め(イタリック)になっていることがありますよね。斜めにするには<I>と</I>で囲みます。ただし、日本語で斜めにすると結構見にくいものですので、気をつけましょう。

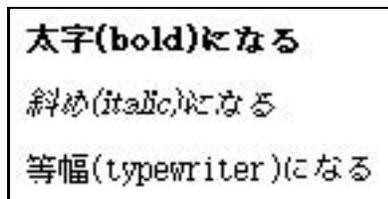
<TT>と</TT>

さらに、タイプライターのような文字(いわゆる「等幅(固定幅)フォント」)を使いたいこともありますよね。特に、計算機のプログラムを載せたい場合や文字だけで絵を表現したい場合には、必要でしょう。そのためには、<TT>と</TT>で囲みます。ただし、もしも1行を越えるような長い場合には、8.4.13で述べる方法を使ったほうが良いでしょう。

入 力

```
<B>太字 (bold) になる</B>  
<I>斜め (italic) になる</I>  
<TT>等幅 (typewriter) になる</TT>
```

↓



太字(bold)になる  
斜め(italic)になる  
等幅(typewriter)になる

図 8.7 文字の修飾

### 8.4.11 リスト(箇条書き)

HTMLでリストにする方法には、3種類あります。すなわち、1) 記号付きリスト、2) 番号付きリスト、3) 見出し付きリストです<sup>22</sup>。

#### 記号付きリスト

記号付きのリストを作るには、<UL>と</UL>で囲み、それぞれの項目の前に<LI>を付けます。

<sup>22</sup> それぞれ、 $\text{\LaTeX}$  の 1) `itemize` 環境、2) `enumerate` 環境、3) `description` 環境にほぼ対応すると考えればよいでしょう。

入 力

記号付きリスト

```
<UL>
  <LI> First Call
  <LI> Glad
  <LI> 2nd Chapter of Acts
</UL>
```



- First Call
- Glad
- 2nd Chapter of Acts

図 8.8 記号付きリスト

番号付きリスト

番号付きのリストを作るには、<OL>と</OL>で囲み、それぞれの項目の前に<LI>を付けます。

入 力

番号付きリスト

```
<OL>
  <LI> David Grisman
  <LI> Tony Rice
  <LI> Doc Watson
</OL>
```



1. David Grisman
2. Tony Rice
3. Doc Watson

図 8.9 番号付きリスト

見出し付きリスト

見出し付きのリストを作るには、<DL>と</DL>で囲み、見出しの前に<DT>を、説明の前に<DD>を付けます。それぞれの項目の前に<LI>を付けます。

入 力

見出し付きリスト

<DL>

<DT> First Call

<DD> CCM界のコーラスグループの第一人者。ロックからジャズコーラスまでこなす幅広い音楽性が魅力。代表作は、Stevie Wonderの"Don't You Worry 'Bout A Thing"などを収録した<I>Human Song</I> (1992)。

<DT> Glad

<DD> 複雑なアカペラもこなす TAKE6 と並ぶ CCM界の有数のコーラスグループ。古典的な hymn を現代的な厚いアカペラにアレンジするその手腕は見事。代表作は、<I>THE ACAPPELLA PROJECT</I> (1988)。

</DL>



First Call

CCM界のコーラスグループの第一人者。ロックからジャズコーラスまでこなす幅広い音楽性が魅力。代表作は、Stevie Wonderの"Don't You Worry 'Bout A Thing"などを収録した*Human Song* (1992)。

Glad

複雑なアカペラもこなす TAKE6 と並ぶ CCM界の有数のコーラスグループ。古典的な hymn を現代的な厚いアカペラにアレンジするその手腕は見事。代表作は、*THE ACAPPELLA PROJECT* (1988)。

図 8.10 見出し付きリスト

## リストの入れ子

これらのリストを組み合わせ、入れ子にすることもできます。

入 力

```
<OL>
  <LI> CCM
    <DL>
      <DT> First Call
      <DD> CCM界のコーラスグループの第一人者。
      <DT> Glad
      <DD> 複雑なアカペラもこなす CCM 界の有数のコーラスグループ。
    </DL>
  <LI> Bluegrass
    <UL>
      <LI> David Grisman
      <LI> Tony Rice
      <LI> Doc Watson
    </UL>
</OL>
```

↓

```
1. CCM
   First Call
     CCM界のコーラスグループの第一人者。
   Glad
     複雑なアカペラもこなす CCM 界の有数のコーラスグループ。
2. Bluegrass
   ○ David Grisman
   ○ Tony Rice
   ○ Doc Watson
```

図 8.11 リストの入れ子

#### 8.4.12 字下げ(インデント)する

本などから引用する場合、行を少しだけ頭とお尻を引っ込ませて、真ん中に集まるような形になっていることがあります。このように、行の頭を引っ込ませることを「インデント」とよびます。インデントさせたい部分を<BLOCKQUOTE>と</BLOCKQUOTE>で囲みます。

入 力

```
これが普通。  
<BLOCKQUOTE>  
こうすると、頭とお尻が引っ込む。  
</BLOCKQUOTE>
```

↓

```
これが普通。  
    こうすると、頭とお尻が引っ込む。
```

図 8.12 字下げ(インデント)する

#### 8.4.13 書いたものをそのまま表示

どのブラウザでもきちんと表示される表、コンピュータのプログラム、アルファベットを使って書いた絵などを表示したい場合には、そのまま書いただけではいくら改行したり空白を空けてもうまくいかないことが多いでしょう。また、フォントもタイプライターのような等幅(固定幅)フォントでないと都合が悪いです。そういう場合には、書いたものをそのまま表示してくれる<PRE>と</PRE>で囲みます。このタグで囲まれると、等幅フォントが使われて、空白や改行がそのまま表示されます。

入 力

```
<PRE>  
import java.applet.* ;  
import java.awt.* ;  
public class JavaWorld extends Applet {  
    public void paint(Graphics g) {  
        g.drawString("Hello Java World",20,30) ;  
    }  
}  
</PRE>
```

↓

```
import java.applet.* ;
import java.awt.* ;
public class JavaWorld extends Applet {
    public void paint(Graphics g) {
        g.drawString("Hello Java World",20,30) ;
    }
}
```

ただし、気を付けなければいけないのは、そのまま表示されるとはいても、その中に例えば<P>のようにタグを入れると、それがそのまま表示されずに、タグと解釈されるということです<sup>23</sup>。<P>の場合でしたら、改行して1行空けてしまいます。もし、<PRE>と</PRE>の中で、例えば<P>を表示したければ、8.4.4に記した方法で書く必要があります。つまり、<P>ならば、

入 力

```
&lt;P>
```

と書かなければなりません<sup>24</sup>。

#### 8.4.14 画像を入れよう (インライン画像)

WWW ページを作っていると、文字だけではちょっと寂しくなってくると思います。自分の写真や絵を入れたい！と思うこともあるでしょう。最近の WWW ページでは、論文などだけを載せているのではない限り、ほとんど絵とか写真が入って華やかになっているものです<sup>25</sup>。

WWW ページを表示する時に同時に読み込まれる画像のことを、「インライン画像」と言います。ここでは、このインライン画像の入れ方について見てみましょう。

##### 画像を用意する

まずは、WWW ページに入れる画像を用意しなければなりません。それについては、ここで詳しく説明する余裕は残念ながらありません。例えば、写真や絵をスキャナと呼ばれる機械で取り込んだり、最近流行のデジタルカメラで撮った写真を取り込んだり、コンピュータを使って書いた絵を使ったり<sup>26</sup>、という様々な方法が考えられます。ここでは、そうした画像は既に用意されているという前提で話を進めます。

大切なことは、入れたい画像は GIF (じふ) というフォーマットで保存されていて、ファイル名の最後に.gif という拡張子を持っていないといけない、ということです<sup>27</sup>。そして、画像のサイズはあまり大きくしないようにしましょう。大きな画像はどうしてもファイルサイズも大きくなりがちで、表示されるのに時間がかかるようになってしまいます。その結果、その画像が入ったページへのアクセスが敬遠されることにもなりかねません。

<sup>23</sup> この点、 $\text{\LaTeX}$  の verbatim 環境は、ほとんどそのままを表示してくれます。

<sup>24</sup> ちょっと面倒ですね。しかし、cc 環境の Mule なら、8.6.1 に記す html-helper-mode を使うことによって、簡単に入力できます。例えば、&amp; を入力するのは、C-c & でできます。

<sup>25</sup> 論文しか書かれていないページですらも、どこかに画像が入っていたりします。:-p

<sup>26</sup> 例えば、cc 環境では X 環境で、xv や xpaint という作画ツール使うことができます。

<sup>27</sup> 少し前まではインライン画像で使えるのは GIF ファイルだけでしたが、256 色までしか使えず、写真の表示はどちらかという苦手という制約がありました。最近では JPEG (じえいべく) と呼ばれる 256 色以上使えるより高品位の、主に写真を表示するための画像も使えるようになってきました (拡張子には .jpeg または .jpg などが付いています)。しかし、色数が多ければファイルサイズは大きくなるので、注意しましょう。白黒も意外に効果的ですよ。

タグの書き方は？

このような GIF フォーマットの画像を、自分の WWW ページがあるディレクトリに置きます。仮に、ファイル名を `mystudio.gif` としておきます。そして、WWW ページの中で画像を入れたいところを決めておきます。なお、画像も文字と同じように扱われますので、`<BR>`などを上手に使いましょう。

タグの書き方は、`<IMG SRC=" " ALT=" ">`です。の部分には、先ほどの画像のファイル名の `mystudio.gif`を入れます。の部分には、その画像を表すタイトルのようなものを入れておきます。これは、もしブラウザでも何かの事情で画像が表示できなかったり、画像を表示しないブラウザを使った時に、それがどういう画像なのかを文字で表示してくれるようにするためです。全体では、次のようになるでしょう。

入 力

```
<IMG SRC="mystudio.gif" ALT="私のスタジオ">
```

図 8.13 のように画像が表示されれば、問題はありません。



図 8.13 インライン画像 (本当はもっときれいに表示されます) :-p

もし、画像が表示されない場合には、図 8.14 のように `ALT=""` に書いた文字が表示されます。

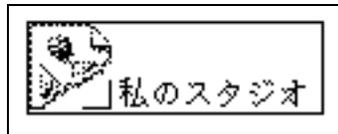


図 8.14 画像が表示されない時

画像が WWW ページと違うディレクトリにある時には、ディレクトリ名を画像ファイル名の前に付けなければなりません。詳しくは、後の 8.4.15 で述べるリンクの張りかたと同じように、ディレクトリを指定します。仮に、WWW ページと同じディレクトリにある `photos` というディレクトリの中にあるとすれば、

入 力

```
<IMG SRC="photos/mystudio.gif" ALT="私のスタジオ">
```

のように書きます。

## 8.4.15 リンクを張ろう

このリンクこそが、HTMLの本領です！何と言っても、リンクできるからこそ「ハイパーなテキスト」なんです。これを使うことによって、世界のどこの WWW ページにも一瞬で<sup>28</sup>つながります。要するに、自分用の「どこでもドア」みたいなものですね。:-)

リンクを張るためには、つなげたいリンク先の URI が分かっている必要があります。タグは、`<A HREF=" " ></A>`というのを使います (A は「アンカー (錨)」の略です)。`"`には、リンク先の URI が入ります。`<A HREF=" " ></A>`の間には、そのページの名前などを入れます。これがないとリンクするポイントができないので、忘れないようにしましょう！

リンクは、ブラウザで見ているときにはその部分をクリックするだけで良いので、ピンと来ないかもしれません。けれども、いざ書く時にはいくつか種類があります。ここでは、

1. 同じサイト内の別のページへのリンク、
2. ページの中の特定の箇所へのリンク、
3. 別のサイトへのリンク

について説明しましょう。

### 同じサイト内の別のページにリンク

この場合、2種類考えられるでしょう。

1. 同じディレクトリにあるファイルへのリンク、
2. 別のディレクトリにあるファイルへのリンク

です。これらのことをするためには、ディレクトリのことがある程度分かっていないと、しんどいかもしれません。分からなければ、「ファイルの階層構造」の部分などを読みましょう。

同じディレクトリにある別のファイルにリンクを張るのは、簡単です。次のようにファイル名を指定するだけで OK です。

入 力

```
<A HREF="atake.html">あたけ君のページ</A>
```

このように書くと、ブラウザで表示される時には通常色が変わられて (デフォルトは青です)、下線が引かれることもあり<sup>29</sup>、マウスの矢印 (カーソル) を合わせるとウインドウの下の部分に URI が出てきて、リンクしていることを示すようになります。

もし、別のディレクトリにあるファイルにリンクを張りたい場合には、ディレクトリ名を付け加えれば OK です。例えば、WWW ページのファイルがあるディレクトリに `music` というディレクトリがあって、その `music` ディレクトリの中の `favorite.html` というファイルにリンクしたいのであれば、次のように書きます。

入 力

```
<A HREF="music/favorite.html">お気に入りの音楽</A>
```

<sup>28</sup> というのは大げさですが。:-) 最近の WWW ユーザの飛躍的な増加に回線増設がなかなか追いつかないこともあって、以前ならすぐにつながるページでも、時間がかかるようになってきました。

<sup>29</sup> Mac 版の Netscape では、基本的には下線が付きません。

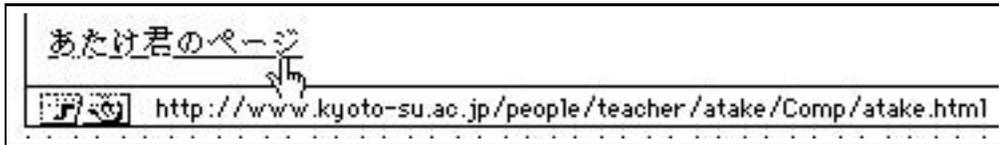


図 8.15 リンクのタグが付けられた文字列

また、リンク先には WWW ページ以外のものを指定することもできます。例えば、ブラウザで画面のある部分をクリックしたら、画像が出てくるようにすることもできます。例えば、画像ファイルで atake.gif にリンクを張るのであれば、

```
<A HREF="atake.gif">あたけ君の写真</A>
```

のように指定しておけば、「あたけ君の写真」という文字列にリンクのポイントができますので、ブラウザでそこをクリックした時に atake.gif の画像を表示させることができます。音声ファイルや動画ファイルなども同じように指定することができます<sup>30</sup>。

#### ページの中の特定の箇所にリンク

ページの中の特定の箇所、例えば同じページの中の別のところにリンクを張るには、ちょっと工夫が必要です。まず、リンク先にあらかじめ印を付けておかなければなりません。例えば、そのページの中の「CCM」という文字列にリンクを張りたいのであれば、それを<A NAME=" " ></A>で囲みます。そして、には何か名前をつけておきます。なお、このタグは<A NAME=" " >で、<A HREF=" " >とは微妙に違いますので、注意してください。

そして、リンク元では、<A NAME=" " >の であらかじめ指定した名前を、<A HREF=" " >の に、頭に#を付けて書きます。そのようにして、リンク先を指定します。

入 力

```
<A NAME="ccm">ここが CCM のコーナー</A>です。  
<A HREF="#ccm"><P>CCM のコーナー</A>にリンクします。
```

↓

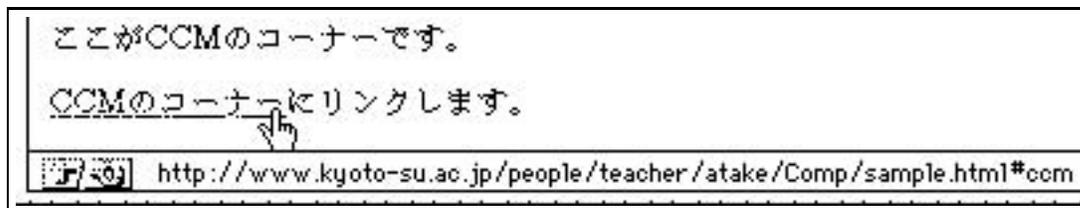


図 8.16 同じページの中でリンク

これで、色が変わっていたり下線が引かれている「CCMのコーナー」の部分をクリックすれば、上の「CCMのコーナー」にジャンプします。

<sup>30</sup> 音声ファイルや動画ファイルの扱いについては、巻末の HTML の参考文献をご覧ください。

繰り返しますが、`<A NAME=" " ></A>`であらかじめリンク先を作っておかないと、同じページ内のリンクは機能しないので気をつけましょう<sup>31</sup>。

## 別のサイトにリンク

別のサイトにリンクする場合には、`<A HREF=" " >`の に URI を入れます。例えば、ホワイトハウスの URI は `http://www.whitehouse.gov/` なので、これを に入れて、`<A HREF=" " ></A>`の間に何か文字などを入れます。

入 力

```
<A HREF="http://www.whitehouse.gov/">ホワイトハウス</A>
```

また、ここでもリンク先は WWW ページに限りません。よく見られるのは、様々なファイルが蓄積されている「FTP サイト」にリンクが張ってある場合でしょう。例えば、京都産業大学の FTP サイトにリンクを張るのであれば、次のようにタグを書きます<sup>32</sup>。

入 力

```
<A HREF="ftp://ccftp.kyoto-su.ac.jp/">京都産業大学の FTP サイト：ファイルがいっぱい</A>
```

あとは、これらを組み合わせれば、世界中のどのページともリンクすることができます！よく行くページのリンクを集めた「リンクのページ」を作っている人も多いです<sup>33</sup>。しかし、リンクのページはいわば借り物を集めたページなので、それ以外にも自分のオリジナルのページを作りましょう。:-)



これで、基本編は終了です。ともかく、色々トライしてみましよう。:-)

<sup>31</sup> 例外として、そのページの一番先頭に戻るようしてくれる `<A HREF="#top"></A>` というタグは、`<A NAME=" " ></A>` のタグでリンク先を指定しなくても機能します。

<sup>32</sup> 以下の URI は変更されるかもしれませんが、もちろん事前にアナウンスはされますが、ネットニュースの `sandai.comp.announce` あたりをチェックしておくとい良いでしょう。

<sup>33</sup> リンク集のページを見ると、その人がどういうことに興味があるのかが分かってなかなか面白いものです。

## 8.5 HTML もっともっと

これまでの「基本編」では、どのブラウザでもほとんど問題なく表示できる基本的なタグを紹介したつもりです。しかしながら、ちょっとカッコいいなあと思うページを見てそのファイルの中身を見てみると、上に記した以外のタグがたくさん使われていると思います。それらを全部紹介することはとてもできませんが、ここではその中でも(あまりマシンに負荷をかけずに)多少役に立つかもしれないものを紹介しておきます。その前に、次にのことにちょっと気を付けるといいでしょう。

### 8.5.1 他の人はうまく見える?

この後に挙げるタグを使うと、確かにちょっと見栄えが良くなるかもしれませんが、世の中には今あなたが使っているブラウザ(例えば Netscape Navigator 4.04)を使って見ているとは限らない場合があります。あなたの目には完璧で美しく見えたとしても、別の人が別のブラウザを使っていると全く違うように見えてしまって、極端な場合あなたの意図していたように見えないどころかぐちゃぐちゃに見えてしまう可能性があります<sup>34</sup>。それだけでなく、最悪の場合、タグによってはそれに対応していないブラウザ(例えば Mosaic など)が強制的に終了してしまうこともあります<sup>35</sup>。

産大で使われているブラウザは殆どが Netscape Navigator ですし、現在世界の多くで使われているブラウザは Netscape Navigator か MSIE であるのは確かなので、以下に挙げるタグを使っても Netscape2.0 以上ならば問題はあまりないのかもしれませんが、しかし、もしそれらのタグや、巻末の参考文献に載っているマルチメディアを意識したタグ、Java などのインタラクティブな要素を盛り込んだタグを使う場合には、そのページの中に「このページは Netscape Navigator 2.0 以上でご覧ください」というようなことを書いておくことも一つの方法です。ネットワークの世界では(現実もそうですが)、自分が見たり受け取ったりする立場になったらどう思うか、をいつも考えておくのがうまくコミュニケーションをするコツです。:-)

### 8.5.2 背景、本文などの色を指定する

ブラウザでページを見ると、指定していない限り基本的に背景の色はグレイで本文の色は黒だと思えます。しかしながら、文字の色が変えられるのと同様に、背景や本文全体の色も変えることができます。例えば、背景の色を白に変えることによって全体を明るい感じにし、さらに文字とのコントラストをはっきりさせることができます。

背景や本文全体の色を変更するには、ソースの最初の方に現れる基本的なタグ<BODY>タグの中に少し書き足すことによって可能になります。背景の色であれば、<BGCOLOR="#??????">を書き加えます。#??????の部分の?マークは、左から順に2桁ずつ赤、緑、青の度合いを表しており、16進法(0123456789abcdef)を使って指定します。例えば、黒ならば#000000、白ならば#ffffffと指定します。少し細かく言うと、赤の場合ならば、#ff0000で真っ赤に、中間の#550000だと、ちょっと暗い赤になります。つまり、数字が大きくなれば色は明るく、小さくなれば暗くなります<sup>36</sup>。背景の色のほかに、以下の色が変わります。

<sup>34</sup> 例えば、8.5.8 に記す「表」などはその可能性が大きいです。

<sup>35</sup> その道の用語で「死ぬ」「落ちる」などと言います。:-p

<sup>36</sup> 16進法で書くのは、ちょっと面倒なのも事実です。<FONT COLOR="black">黒</FONT>のように色の名前を書いて指定することもできる場合もありますが、どの色でも表せるわけではなくて基本的な16色(black, green, silver, lime, gray, olive, white, yellow, maroon, navy, red, blue, purple, teal, fuchsia, aqua)に限定されます。ですので、中間色を表現したい場合には、16進法での表現の仕方を研究してください。

項目	タグの書き方	(16進数指定の)例
背景の色	BGCOLOR="#??????"	BGCOLOR="#ffffff"
本文全体の色	TEXT="#??????"	TEXT="#000000"
まだ訪れていないリンクの色	LINK="#??????"	LINK="#ff0000"
既に訪れたリンクの色	VLINK="#??????"	VLINK="#0000ff"
リンクをクリックした瞬間に変化する色	ALINK="#??????"	ALINK="#ff0000"

これらを実際書くと、以下のようになります。

入 力

```
<BODY BGCOLOR="#ffffff" TEXT="#000000" LINK="#ff0000" VLINK="#0000ff"
ALINK="#ff0000">
そして本文を書くのであった。
</BODY>
```

### 8.5.3 文字の色を変える

ブラウザでは、背景や本文全体のほかに、1文字ごとに色を自由に変えて表示することができます。その場合<FONT COLOR="#??????">と</FONT>で囲みます。色の指定の仕方は、8.5.2と同じなので参照してください。以下に、代表的な色だけを記しておきます。

入 力

```
<FONT COLOR="#000000">黒</FONT>
<FONT COLOR="#ff0000">赤</FONT>
<FONT COLOR="#00ff00">緑</FONT>
<FONT COLOR="#0000ff">青</FONT>
```

気を付けておかなければいけないのは、背景の色とのコントラストです<sup>37</sup>。背景と文字を似たような色にしてしまうと、非常に見にくいページになってしまいますので、気をつけましょう<sup>38</sup>。

また、1文字ごとに色を変えたりすると、文字の間に入るタグのせいで検索する時にうまくいなくなることがありますから、特殊な効果を狙わない限りやめておいた方が無難です。また、ページ全体の文字の色を変えたい場合には、この方法ではなくて先に述べた8.5.2の方法を使うと良いでしょう。

### 8.5.4 見出しの位置を変える

8.4.9に述べた<Hn></Hn>を使って指定をする見出しは、特に指定しないと左寄せになります。しかし、真ん中や(あまりないかもしれませんが)右に寄せたい場合があると思います。その場合には、<Hn>タグの中にALIGN=" "という項目を追加して、の部分に、centerやrightを指定します。例えば、一番大きな見出しである<H1>タグを真ん中寄せにしたい場合には<sup>39</sup>、

入 力

```
<H1 ALIGN="center">JavaScriptのページ</H1>
```

<sup>37</sup> 背景の色に関しては、8.5.2を参考にしてください。

<sup>38</sup> 見にくい一例として、背景を黒にして文字を白に指定してしまうと、ブラウザによっては文字が全く見えなくなってしまいます。文字を白に指定するのは避けた方が良いでしょう。

<sup>39</sup> 真ん中寄せにする別の方法については、8.5.7を見てください。

また、<H5>で右寄せにしたい場合には次のように書きます<sup>40</sup>。

入 力

```
<H5 ALIGN="right">HTML をよりインタラクティブに</H5>
```

↓

## JavaScript のページ

HTML をよりインタラクティブに

### 8.5.5 段落の位置を変える

8.4.7 に述べた<P>を使って段落を区切る場合、特に指定しないと左寄せになります。しかし、真ん中や右に寄せたい場合があると思います。その場合には、8.4.9 で述べたのと同じように<P>タグの中に ALIGN="center"あるいは ALIGN="right"という項目を追加します<sup>41</sup>。

入 力

```
<P ALIGN="center">これで真ん中寄せ
```

```
<P ALIGN="right">これで右寄せ
```

↓

これで真ん中寄せ

これで右寄せ

### 8.5.6 特定の文字の大きさを変える<sup>42</sup>

ある特定の文字だけ大きさを変えるにはいくつか方法がありますが、ここでは簡単な方法を記します。例えば、文字を大きくしたい場合にはその文字を<BIG></BIG>で、小さくしたいときには<SMALL></SMALL>で囲みます<sup>43</sup>。さらに、上付きにしたいときには<SUP></SUP>で、下付きにしたいときには<SUB></SUB>で囲みます。

入 力

```
標準 <SMALL>小さい文字</SMALL> <BIG>大きい文字</BIG>
```

```
めちゃ<SUP>2</SUP>いけてる 水はH<SUB>2</SUB>O
```

<sup>40</sup> 右寄せにする別の方法については、8.5.7 を見てください。

<sup>41</sup> 真ん中寄せ、右寄せにする別の方法については 8.5.7 を見てください。

<sup>42</sup> ただし、cc 環境の UNIX コンピュータの Netscape の日本語フォントでは対応していない場合があります。

<sup>43</sup> その他の方法としては、大きさを変更したい文字を<FONT SIZE="n"></FONT>で囲みます。n には 1 から 7 までの数字が入ります。7 が最も大きくて、1 が最も小さい文字です。注意しなければならないのは、<Hn>と</Hn>とは逆に大きい数字の時に大きいフォントになるということです。すなわち、

```
<FONT SIZE="7">一番大きい文字</FONT>
```

```
<FONT SIZE="1">一番小さな文字</FONT>
```

ですが、この方法にはちょっと慣れが必要なようです。

↓

標準	小さい文字	大きい文字
めちや <sup>2</sup>	いけてる	水は H <sub>2</sub> O

### 8.5.7 真ん中寄せ、右寄せにする

ページを作成していると、ある文字列を行の中央にしたいとか、名前や日付などは右に寄せたいとかいうこともあると思います。

文字を真ん中に寄せるには、<CENTER>タグを使って次のように囲みます<sup>44</sup>。

入力

```
<CENTER>真ん中寄せにする</CENTER>
```

文字列を右寄せにしたい場合には、<DIV>というタグを使い、そのタグの中に ALIGN="right" を指定して、文字列を囲みます。

入力

```
<DIV ALIGN="right">右寄せにする</DIV>
```

↓

真ん中寄せにする	右寄せにする
----------	--------

図 8.17 真ん中寄せと右寄せ

### 8.5.8 簡単な表を作る

HTML では、表を作ることができます。それも、かなり複雑な表ができます。しかし、かなり面倒でもあります。まずは簡単な表を作ってみますが、それでもいくつかのタグが必要になります。

<TABLE>と</TABLE>

表全体は、<TABLE BORDER>と</TABLE>で囲みます。<TABLE>タグの中の BORDER は、枠線を引くための命令です。枠線が必要なければ、<TABLE>だけで OK です。

<TR>と</TR>

<TR>と</TR>の間には、横 1 列に入る要素を書きます。

<sup>44</sup> 真ん中寄せには、次の右寄せと同じような<DIV ALIGN="center">と</DIV>で囲む書き方もあります。

<TD>と</TD>

<TD>と</TD>の間には、個々のマス(「セル」と言います)に入る要素を書きます。

<TH>と</TH>

<TH>と</TH>の間には、個々のセルに入る要素を書きますが、見出し(ヘッダ)になりますので、太字(bold)になって、真ん中寄せになります。

<CAPTION>と</CAPTION>

<CAPTION>と</CAPTION>の間には、表の表題を書きます。何も指定しないと(つまり、「デフォルト」では)表の上に、真ん中寄せで表示されます。

ここまでで、基本的な表が作れるようになります。非常に簡単な時間割を作ってみましょう。

入 力

```
<TABLE BORDER>
<TR>
<TH></TH> <TH>月</TH> <TH>火</TH>
</TR>
<TR>
<TH>2</TH> <TD>ヘブライ語 I</TD> <TD>言語学セミナー A</TD>
</TR>
<CAPTION>簡単な時間割</CAPTION>
</TABLE>
```

↓

	月	火
2	ヘブライ語I	言語学セミナーA

図 8.18 Netscape の表

もう少し複雑な時間割を作ってみましょう。

## 入力

```

<TABLE BORDER>
<TR>
<TH></TH> <TH>月</TH> <TH>火</TH> <TH>水</TH> <TH>木</TH> <TH>金</TH> </TR>
<TR>
<TH> 2 </TH> <TD>ヘブライ語 I</TD> <TD><BLINK>言語学セミナー A</BLINK></TD> <TD>卒論指導</TD>
<TD>卒論指導</TD> <TD></TD>
</TR>
<TR>
<TH> 3 </TH> <TD></TD> <TD>言語学演習 B</TD> <TD></TD> <TD>卒論指導</TD> <TD></TD>
</TR>
<TR>
<TH> 4 </TH> <TD>言語学原論</TD> <TD></TD> <TD></TD> <TD></TD> <TD>ドイツ語 I</TD>
</TR>
<TR>
<TH> 5 </TH> <TD></TD> <TD>ヘブライ語 II</TD> <TD></TD> <TD></TD> <TD></TD>
</TR>
<CAPTION>ある先生のある年度の時間割 ;-></CAPTION>
</TABLE>

```

↓

	月	火	水	木	金
2	ヘブライ語I	言語学セミナーA	卒論指導	卒論指導	
3		言語学演習B		卒論指導	
4	言語学原論				ドイツ語I
5		ヘブライ語II			

図 8.19 ちょっと複雑な表

HTML で表を書く場合、最初はなかなかうまくいかないものです。ただ、ソースをいかに見やすく書くかによって、修正がしやすくもしくもなります。焦らずに根気よくトライして、自分なりの見やすい方法を開発しましょう。

## 8.6 WWW ページ作成に役立つもの

### 8.6.1 HTML を書くお助けツール：html-helper-mode

さて、これまで HTML の書き方を説明してきましたが、注のところでたびたび「html-helper-mode を使うと便利やでー」と書いてきました。このような便利なものが、cc 環境の UNIX コンピュータの Mule

には備えられています。例えば、リンクを張りたい時に、`<A HREF="http://www.whitehouse.gov/">`ホワイトハウス`</A>`というのを間違いなく書くのは、なかなか神経を使いますよね。ところが、`html-helper-mode` を使えば、`C-c C-a l`<sup>45</sup> と押すだけで、自動的に`<A HREF=""></A>`を挿入してくれます。それだけではなく、カーソルも 2 番目の”の上に置かれるので、すぐに URI を書き込むことができます。

`html-helper-mode` は、基本的な HTML のタグには対応しています<sup>46</sup>。基本的とは言っても、実はここで紹介したよりもはるかに多くのタグがあります。そして、どのタグがどのキーに割り当てられているかを覚えるのはちょっと大変かもしれません。「こんな覚えるのはかなん」というあなたは、京都産業大学の WWW ページからたどれる「各種ドキュメント」の中に「`html-helper-mode` のキー割り当て一覧」が載せられていますので、それを見るなり印刷して持ち歩くなりすると良いでしょう<sup>47</sup>。

次の表 8.1 では、そのキー一覧割り当て表の中から、ここで紹介したタグのキー割り当てを記します (元の表の中の項目とは順番を変えて、説明を書き足しています)。

## 8.6.2 HTML 文法チェッカ : `jweblint`

WWW ページを書き始めの頃はどのように書いていいのか、これで正しいのかどうか分からないものですね。けれども、Netscape などの一部のブラウザは HTML がある程度いい加減でもそれなりの表示してくれます。このように、ある程度のミスは許容されるという懐の深さが利点と言えるでしょうけれども、逆に自分のミスに気づきにくいという欠点でもあります。

しかし、どのブラウザでもきちんと見えるページを作る場合には、いい加減に HTML を書いているとうまく表示されないだけでなく、ブラウザを強制的に終了させてしまいかねません。自分の書いた HTML が正しいかどうかを判断できると良いですね。

`cc` 環境では、自分が書いたページの HTML をチェックすることができます。使い方は非常に簡単です。例えば、自分のホームページの文法をチェックしたい場合には、自分のホームページのありか (ホームディレクトリの中の `public.html` ディレクトリ) へ移動して次のコマンドを打ってください。

```
cc2000(9)% jweblint index-j.html
```

とあるページにかけてみたら以下の結果が出ました (なお、Netscape 拡張を許容するようにオプションを追加しました<sup>48</sup>)。

```
cc2000(10)% jweblint -x Netscape index-j.html
index-j.html(11):  組合わせエレメント <P> が空です.
index-j.html(23):  </A> がつり合いません (マッチする <A> が見つかりません).
index-j.html(39):  組合わせエレメント <P> が空です.
```

もちろん、他のページの文法もチェックできます。`index-j.html` の部分をチェックしたいファイル名に変えてください。

---

<sup>45</sup> アルファベットの L の小文字です。

<sup>46</sup> 1997 年 3 月時点では、上記のいくつかのタグには完全に対応していないようです。対応していないタグについては、自分でガリガリと書くか EGG で単語登録をするのも一つの方法です。

<sup>47</sup> URI は <http://www.kyoto-su.ac.jp/information/html-helper-mode/keybindings217.html> です。

<sup>48</sup> このオプションを付けないと、以下のようなエラーを大量に表示してくれるもので...

```
index-j.html(6):  <body> の 'BGOLOR' 属性は拡張マークアップです (これを許すには "-x <extension>" を使ってください).
index-j.html(7):  <center> は 拡張マークアップです (これを許すには "-x <extension>" を使ってください).
index-j.html(19):  <IMG> の 'BORDER' 属性は拡張マークアップです (これを許すには "-x <extension>" を使ってください).
index-j.html(28):  </center> は 拡張マークアップです (これを許すには "-x <extension>" を使ってください).
```

基本キー	機能	押すキー	挿入されるタグ
C-c C-b	head element(begin の b)	C-c C-b t	<TITLE></TITLE>
	特殊文字	C-c > C-c < C-c &	&GT; &LT; &AMP;
	基本	M-TAB M-RET C-c RET C-c - M-C-t	タグの補完 <P>   <HR> <!-- HHMTS START --> セーブ時に timestamp が入ります <!-- HHMTS END -->
C-c C-t	ヘッダ (title の t)	C-c C-t 1 C-c C-t 2 C-c C-t 3 C-c C-t 4 C-c C-t 5 C-c C-t 6	<H1></H1> <H2></H2> <H3></H3> <H4></H4> <H5></H5> <H6></H6>
C-c C-p	文字の修飾 (物理的指定): physical style (physical の p)	C-c C-p b C-c C-p i C-c C-p f	<B></B> <I></I> <TT></TT>
C-c C-s	文字の修飾 (論理的指定): logical style (style の s)	C-c C-s a C-c C-s b C-c C-s p	<ADDRESS></ADDRESS> <BLOCKQUOTE></BLOCKQUOTE> <PRE></PRE>
C-c C-l	リスト (list の l)	C-c C-l d C-c C-l u C-c C-l o C-c C-l i C-c C-l t C-c C-l l	<DL><DT><DD></DL> <UL><LI></UL> <OL><LI></OL> <LI>か<DT><DD>の適当と思われる方 <DT><DD> <LI>
C-c C-i	イメージ (image の i) C-i は Tab キーに相当	C-c C-i t	<IMG ALT="" SRC="">
C-c C-a	アンカー (anchor の a)	C-c C-a l C-c C-a n	<A HREF=""></A> <A NAME=""></A>

表 8.1 本文で紹介したタグと html-helper-mode のキーバインディング

## 8.7 HTMLのこれから

### 8.7.1 最新の規格：HTML4.0

これまで、どのブラウザで表示してもほぼ同じ表示になる基本的なタグを紹介してきました。これは少し前のHTML2.0およびHTML3.2という規格に基づいています。一方、8.4で紹介したように、現在の最も新しい規格はHTML4.0です。この規格は、現在の最新のブラウザであるNetscape4.0およびMSIE4.0で既にサポートされているタグをかなり取り入れています(取り入れられていないタグもたくさんありますけれども)。

ここで少しHTML4.0の大まかな特徴を紹介しておきましょう(リメイ [6, p.612] に若干追加しました。追加部分は [] で示しています)。詳しくは

<http://www.w3.org/WWW/TR/>

をご覧ください。

- 国際化 — 文字集合にISO 10646標準規格であるUCS (Universal Character Set、現時点ではUCS-2と呼ばれる文字集合はUNICODEと同じ)を採用し、各国語のHTMLドキュメント作成を正式にサポートした。
- アクセシビリティ — (ハンディキャップのある人もない人も含めて)多様なユーザのために、音声読み上げブラウザのための機能、代替テキスト[つまりALT属性]、長い説明文、各種ラベル情報などを入れる情報などを入れる機能を持たせた。
- 表機能 — 表の内容の構造化や、読み込み途中での表示開始を可能にするための配慮を追加した。
- 複合ドキュメント — イメージやアプレットなどの埋め込みを<OBJECT>タグで統一して扱えるようにした。
- スタイルシート — HTMLによる文書構造の記述と、スタイルシートによる表現の規定を分離した。
- スクリプト機能 — [JavaScriptやVBScriptなどによる]スクリプトによってダイナミックなWebページを作成する土台を提供した。
- 印刷機能 — 単に画面表示と同じものを印刷するのではなく、印刷用ならではの内容制御を行えるようにした。

### 8.7.2 スタイルシートについてちょっとだけ

上記に記したHTML4.0の特徴のうち、「スタイルシート」についてごく少しだけ述べます。これまで、フォントの大きさ、フォントの色、背景の色、配列(右寄せや真ん中寄せなど)をコントロールするためにはいくつかのタグ(基本編で述べた<FONT>や<CENTER>など)が用いられていましたが、HTML4.0ではこれらは「非推奨」(deprecated)となっています<sup>49</sup>。HTML4.0では、HTMLそのものは文書の論理構造(章、節、項、箇条書きなど)の記述に専念することになって、上に述べた表示やレイアウトに関する定義はこのスタイルシート(CSS = Cascading Style Sheet)<sup>50</sup>に担わせていく方向になっています<sup>51</sup>。

<sup>49</sup> これらの多くはHTML3.2で取り入れられたもので、表示やレイアウトに関するタグが大幅に採用されました。しかし、HTML4.0では手のひらを返したように「非推奨」になってしまいました。

<sup>50</sup> 現在「レベル2」(CSS2)まで出ていますが、これをサポートしているブラウザは現時点ではないはずです。早くサポートされることが期待されます。また、某社は独自のスタイルシートを提案していますが、某社のブラウザでないとは正常に表示できない恐れがありますので、ここには記しません。いくら先行されたからといっても...

<sup>51</sup> というよりも、「HTML 4.0はこいつなしには語れないと言っても過言ではない」と言い切る某業界関係者の方もおられます。:-)

例えば、<H1></H1>、<H2></H2>、<H3></H3>などのタグを使うと、8.4.9 で見たようにフォントは太字になって大きさがそれなりに変化します。もしそれを自分の好みにしたい場合には、<FONT></FONT>タグを使っていちいち定義しなければなりません。しかしながら、スタイルシートを使うことによって、自分の好みの定義を一つの文書全体に適用することも可能になります。この場合、<HEAD></HEAD>タグの中に<STYLE TYPE="text/css"></STYLE>タグを挿入して定義します。例えば、<H1></H1>タグを使った場合には 36 ポイントの赤色、<H2></H2>タグを使った場合には 24 ポイントの緑色、<H3></H3>タグを使った場合には 18 ポイントの青色のフォントで表示されるようにしましょう<sup>52</sup>。なお、以下の例は Netscape4.0 以上または MSIE3.0 以上でない则表示できないことをあらかじめお断りしておきます。

#### 入 力

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0//EN">
<HTML>
<HEAD>
<TITLE>An Example of Stylesheet</TITLE>
<STYLE TYPE="text/css"> <!--
H1 font-size:36pt; color:red
H2 font-size:24pt; color:green
H3 font-size:18pt; color:blue
-->
</STYLE>
</HEAD>
<BODY>
<H1>36 ポイントの赤色の超大きな見出し</H1>
<H2>24 ポイントの緑色の非常に大きな見出し</H2>
<H3>18 ポイントの青色の大きな見出し</H3>
</BODY>
</HTML>
```

上の例では一つのページの中ではどれにも適用されますが<sup>53</sup>、自分が作っている全てのページで使いたい場合には、いちいちファイルの最初に書くのも面倒なのでちょっと工夫が必要です。まずはスタイルシートを定義したファイルを作ります。仮にそのファイル名を style01.css として、次のような内容を書きます。

#### 入 力

```
H1 font-size:36pt; color:red
H2 font-size:24pt; color:green
H3 font-size:18pt; color:blue
```

そして、そのスタイルシートを利用したいページの<HEAD>/HEAD>タグの中に次の一行を書きます。

```
<LINK REL="stylesheet" TYPE="text/css" href="style01.css">
```

これで、あなたがスタイルシートで定義したフォントの大きさと色で<H1></H1>タグなどが表示されます<sup>54</sup>。

<sup>52</sup> この本では白黒でしか印刷されませんので、出力結果はご自分でお確かめ下さい。:-)

<sup>53</sup> この方式を Embedding Style Sheet と言います。

<sup>54</sup> この方式を Linking Style Sheet と言いますが、WWW サーバに「text/css」という MIME タイプと「.css」という拡張子が設定されていなければなりません。

さらには、タグの中にスタイルシートを埋め込む方法もあります<sup>55</sup>。一時的に変更したい場合には、手っ取り早いかもしれません<sup>56</sup>。

入 力

```
<H1 STYLE="font-size:36pt; color:red">36ポイントの赤色の超大きな見出し</H1>
<H2 STYLE="font-size:24pt; color:green">24ポイントの緑色の非常に大きな見出し</H2>
<H3 STYLE="font-size:18pt; color:blue">18ポイントの青色の大きな見出し</H3>
```

今後 WWW ページを本格的に作ろうと考えている人は、スタイルシートを研究した方が良いと思います。スタイルシートをサポートしていないブラウザもありますし、Netscape と MSIE ではサポートしている要素が違う場合があります。ですので、現時点ではスタイルシートを使って WWW ページを書くときにも、スタイルシートをサポートしていないブラウザで見てもおかしくないように書く配慮は必要でしょう。

スタイルシートについて詳しいことは、市販の書籍を見ていただくか (例えばマルダー [3])、以下の URI を参考にしてください。

- WWW コンソーシアムの CSS に関するページ — <http://www.w3.org/pub/WWW/Style/>
- クイックリファレンス — [http://webcom.net/~gmc/html/quick\\_ref.html](http://webcom.net/~gmc/html/quick_ref.html)

### 8.7.3 次世代の HTML: XML

これまで紹介した HTML は比較的簡単で覚えやすく、WWW ページが作りやすいために爆発的に普及して、今日の WWW ブームが築かれることになりました。しかしながら、ちょっと凝った細かなことをしようとすると、意外に制約があるのも事実です。何と云っても、HTML は決められたタグしか使うことができません。中には、自分の望む WWW ページを作るために、手軽な JavaScript から本格的な Perl、C などの言語を使ってページを作っている人もいます。

そのような HTML の限界もあって、最近独自のタグを定義でき自分で拡張できる、いわば次世代の HTML が出てきました。それが XML (eXtensible Markup Language) です。XML は今年の 2 月 10 日に W3C (World Wide Web Consortium) によって勧告として制定された、WWW のための新しい文書記述言語です<sup>57</sup>。

一例を挙げてみましょう。以下のようなデータがあるとします。

<sup>55</sup> この方式を **Inline Styles** と言います。

<sup>56</sup> なお、もし Embedding Style Sheet と Linking style Sheet と Inline Styles を同時に定義した場合の優先順位は、  
Inline Styles > Linking Style Sheet > Embedding Style Sheet

です。

<sup>57</sup> XML は SGML (Standard Genralized Markup Language) という米国の国防省や製造業 (例えば、ボーイング社の電子マニュアルが有名どころです) の技術文書制作にも用いられている規格の簡易版で、「SGML Lite」とも言える直系の子どもです。HTML も実は SGML を土台にしているはずなのですが、この放蕩息子は今では遠く離れてかなり独自の道を歩んでしまっているようです。^-^;

## 入 力

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0//EN">
<HTML>
<HEAD>
<TITLE>CD データ : CCM</TITLE>
</HEAD>
<BODY>
<H1>CD データ : CCM</H1>
<UL>
<LI>アーティスト : First Call
<LI>メンバー : Marty McCall, Bonnie Keen
</UL>
<H2>アルバム</H2>
<DL>
<DT>First Call
  <UL>
    <LI>発表年: 1996 年
    <LI>レーベル: WARNER ALLIANCE
    <LI>プロデューサー: Darrel Brown, Cheryl Rogers、コプロデューサー: First Call、エグゼクティブ・プロデューサー: Neal Joseph
    <LI>ゲスト: Veritas, Micael McDonald, Amy Holland, Take 6
    <LI>曲目: Let The Healing Begin, The Prize, Will I Find Love, Selah, This Side Of Paradise, You'll Be There, I Find You, Sunrise On Ararat, Shadow Of The Cross, Breath Of God
  </UL>
</DL>
</BODY>
</HTML>
```

このように書けば、WWW ブラウザではうまく表示はできるでしょうし、印刷も問題はないでしょう。しかしながら、これをデータベースとして利用したい場合、例えば曲目だけを取り出したい、プロデューサーの名前だけを取り出したい時には、困ります。どれも<LI>タグに続いているだけなので、タグによってその中身を区別することはできません。

一方、XML ではそれらを自由に定義できます。

## 入力

```
<?xml version="1.0" encoding="shift-jis"?>
<CD データ : CCM>
<アーティスト>
  <アーティスト名>First Call</アーティスト名>
  <メンバー>Marty McCall, Bonnie Keen</メンバー>
</アーティスト>
<アルバム>
  <アルバム名>First Call</アルバム名>
  <発表年>1996</発表年>
  <レーベル>WARNER ALLIANCE</レーベル>
  <プロデューサ コプロデューサ="First Call" エグゼクティブ・プロデューサ="Neal Joseph">
Darrel Brown, Cheryl Rogers</プロデューサ>
  <ゲスト>Veritas, Micael McDonald, Amy Holland, Take 6</ゲスト>
  <曲目>Let The Healing Begin, The Prize, Will I Find Love, Selah,
This Side Of Paradise, You'll Be There, I Find You, Sunrise On Ararat, Shadow Of
The Cross, Breath Of God</曲目>
</アルバム>
</CD データ : CCM>
```

このように、XML で書くと、曲目だけのデータ、プロデューサだけのデータを取り出してデータベースを作ることができるようになります。

つまり、XML はデータと文書の二面性を持っているとすることができます。XML は HTML のように表示や印刷のためにも使うことができますが、それだけではなくてプログラムが解釈できるようにされています<sup>58</sup>。このことによって、ネットワーク上で動く特定のアプリケーションのために独自のタグを定義することができますので、XML で書かれた文書に埋め込まれた情報を元にして動かすことが可能になります。

このように大変将来性の高い XML ですが、独自のタグを定義できる分それにあつまる様々なもの、例えば文書型定義<sup>59</sup>、リンクの記述<sup>60</sup>、スタイルシート<sup>61</sup>を自分で定義する必要があります。XML 文書を書くためのエディタなどもこれからです<sup>62</sup>、XML を表示してくれるブラウザも現時点では限られていますが、将来性のある XML ですから文書作成やブラウザなども近い将来対応すると思われます。

XML に関する書籍としては村田 [4]、富士通 [2]、倉元 [1] などがあり、雑誌の記事もありますし (例えば、村田 [5])、WWW ページとしては以下のページを紹介しておきます。

- SGML Café — <http://www.fxis.co.jp/DMS/sgml/index.htm>

<sup>58</sup> ちなみに、XML 文書を解析してその構造の一覧 (木構造) を作成し、XML 文書が正当または適正な形式かどうかを検証するものを XML パーサ (parser) と言います。実は MSIE4.0 にはこの XML パーサの機能が既に組み込まれています。つまり、Java で書かれた MSXML in Java や C++ で書かれた MSXML in C++ ですが、両者の検証する対象は少し異なります。Netscape に組み込まれるのも時間の問題と言われていました。

<sup>59</sup> DTD (Document Type Definition) ファイルで、タグの種類や関係および属性を指定するものです。もし DTD ファイルを自動的に生成できるような場合には前もってなくても良いですし、既に公開されている DTD ファイルを利用することもできます。HTML の場合もそれぞれ DTD ファイルがあります。もっとも、DTD ファイルを公開していないために混乱を引き起こしている某社もありますけれども。:-)

<sup>60</sup> XLL (eXtensible Linking Language) で、例えば一つのリンクが 3 つ以上のリソースを持てるなどの HTML よりも進んだ機能を持っています。この原稿の執筆段階では、Xlink と Xpointer という 2 つのパートに分かれました。

<sup>61</sup> XSL (eXtensible Stylesheet Language) で、XML 文書のスタイルシートを記述する言語ですが、まだノートの段階で正式に制定されていません。

<sup>62</sup> Mule にも XML 編集モードを追加することができますが (psgml)、cc 環境の Mule にはまだ入っていないようです。

これまで HTML をとことん使い込んで限界を感じている人は、ぜひ XML にチャレンジして下さい。:-)

## 8.8 おわりに

これで、WWW ページを作るための HTML の基本的な書き方をほぼ一通りの説明してきました。もちろん、これはごくごく基本的なものですし、学内の共用コンピュータで使える Netscape での表示を中心に記してきたので、最近よく使われるようになってきた MSIE 対応の書き方などには触れていません。HTML を含めた WWW の世界は日進月歩です。これ以上のことについては、例えば

<http://www.kyoto-su.ac.jp/information/>

の「HTML 関係」にもいくつか役に立つ情報が載っていますので、まずは見てください。それでも物足りない人は、巻末の HTML の参考文献や書店にあふれている HTML 関係の本を見て、あなたの WWW ページをどんどん改良してください。

あとは、あなたの WWW ページが魅力的になって、たくさんの人が訪れるようになるかどうかは、あなたのセンス次第です。ただ、繰り返しになるかもしれませんが、次のいくつかのことだけは心に留めておいて欲しいのです。

### 全世界の人に見られているんだよ

WWW ページを作るということは、全世界のインターネットのユーザに対してあなたのページを公開するということです。ですので、世界の誰に見られても良い内容にしましょう。全世界に公開されていることから、日本語のページだけではなくて英語のページも作っておくと喜ばれるでしょう<sup>64</sup>。

### 見やすくしようね

誰が見ても、見やすいページにすると良いでしょう。そのためには、<P>や <BR>などでうまく改行したり、<UL></UL>などのリストをうまく活用しましょう。また、</H2>などの / がついた終了タグを書かないというタグの閉じ忘れでその後の部分の表示がおかしくなることが多いので、8.6.2 でも紹介した jweblint でチェックしましょう。

### どのブラウザでもある程度見れるようにしようね

巷では Netscape もしくは MSIE がよく使われており、それらに対応したページが多いのは事実です。仮にそのようなページを作ったとしても、他のブラウザで見てもおかしくないように、複数のブラウザでチェックすると良いでしょう<sup>65</sup>。cc 環境の UNIX コンピュータの X 環境であれば、Netscape の他に Mosaic なども使ってチェックすることができます (今や Mosaic をご存じの方がどのくらいいるのか...:-p)。

また、ディスプレイの大きさによって、全く違うふうに見えるものです。特に、21 教室のワークステーションを使って大きなディスプレイで丁度うまく見えるように作られたページは、例えば 11 教室の Mac

<sup>63</sup> 余談ながらここは「夏期言語学講座」(Summer Institute of Linguistics) という言語学の成果を生かして聖書翻訳を行っているところが運営しており、言語学に関する膨大な情報があります。そうしたことに興味がある方にもぜひお薦めのサイトです。

<sup>64</sup> ちなみに、wwwmkdir を実行してできる index-j.html の j は、japanese の j で、「日本語ページ」という意味です。英語ページを作る場合には、-j をはずして index.html とすると分かりやすいでしょう。

<sup>65</sup> Netscape 自体でもバージョンによって対応するタグが異なっているので、違うように表示されることがあります。

の小さめのディスプレイではそれほどうまく見えなかったりします。ですので、大きさの違うディスプレイを使ってチェックすると良いでしょう。

人の素敵なページを参考にしようね

見やすいページといっても、最初はどういうふうにかいたら良いか分からないかもしれません。ですので、「あっ、これいいなあ」と思ったページに出会ったら、そのファイルの中身を見てどのように書いてあるかを調べて、それを参考にすると良いでしょう<sup>66</sup>。

しかし、いくら参考にするとっても、そのままコピーしたものを使うのではなくて、あなた流のアレンジを加えて取り入れるようにしましょう。そうでないと、ただの「真似しい」になってしまいます。:-)

法律は守りましょう

参考にしたときにはアレンジを、と書きました。また、最初にも書きましたが、自分のオリジナルな情報で勝負してください！このことは、本当にお願ひしたいのです。くれぐれも、雑誌などから写真やアニメ、市販の CD などからの音、市販のビデオからの画像などをあなたの WWW ページに入れることはやめてください！繰り返しますが、もしそういうページが京都産業大学の中で見つかった場合には、ページのオーナーであるあなたに必ず謝罪と削除などのしかるべき対応を求めます。お互いの作品の権利を守って、自分のオリジナルな情報で勝負してください。



これで、HTML 入門はおしまいです。京都産業大学の中で魅力的な WWW ページがますます増えることを期待しています。:-)

---

<sup>66</sup> 個人的には、表、フレーム、JAVA の applet、JavaScript、CGI がそうでした (あとの 4 つはここでは説明していません。巻末の参考文献を参照してください)。

## 参考文献

- [1] 倉元靖史. 1997. 『CD-ROM 付 SGML&XML ガイド』日刊工業新聞.
- [2] 富士通 XML 推進チーム編. 1997. 『はじめての XML』日経 BP 社.
- [3] マルダー, スティーブン著, 安藤慶一訳. 1997. 『Web スタイルシート活用ガイド』プレントイスホール出版.
- [4] 村田真編著. 1998. 『XML 入門』日本経済新聞社.
- [5] 村田真. 1998. 「インターネット最新テクノロジー:16回 次世代の HTML として注目が集まる XML (eXtensible Markup Language)」『インターネットマガジン』40: 350-353.
- [6] リメイ, ローラ, アーマン・ダニッシュ著, 武舎広幸, 久野禎子, 久野靖訳. 1998. 『HTML 入門第 2 版: WWW ページの作成と公開』プレントイスホール出版.

## 第9章 L<sup>A</sup>T<sub>E</sub>X

### 9.1 L<sup>A</sup>T<sub>E</sub>X (らてふ)って何？

#### 9.1.1 L<sup>A</sup>T<sub>E</sub>X とは？

今までワープロを使っていると、数学の式で複雑なものはできませんでした。ところが、L<sup>A</sup>T<sub>E</sub>X はそれをきれいな形で出力できます。また、論文やレポートなどにも手軽に使えます。ただ、ルールがあって、ワープロのようにすぐに使えるわけではありません。また普通のワープロに比べて応用の自由度が大きく、さまざまなことが可能で、絵や図も簡単に文書中に埋め込むことができます。

レポートや卒論をワープロで書こうと思っている諸君、L<sup>A</sup>T<sub>E</sub>X を使って書いてみませんか。

#### 9.1.2 L<sup>A</sup>T<sub>E</sub>X の特徴

L<sup>A</sup>T<sub>E</sub>X の特徴をいくつかあげておきます。

1. 複雑な数式をきれいな形で出力します。したがって、理科系の人にとっては必需品です。(論文など章節のはっきりした構造のある文章を作るのに向いている。)
2. お絵書きツールなどで作った図を簡単にはめ込むことができます。
3. 目次・索引・文献リストを簡単に作ることができます。

#### 9.1.3 L<sup>A</sup>T<sub>E</sub>X の作業の進めかた

大まかに言うと次のようにして進めていきます。

1. エディタ (Mule など) で文書ファイル<sup>1</sup> を作成する。
2. 作成したファイルを L<sup>A</sup>T<sub>E</sub>X で処理 (コンパイル) する。
3. xdvi でコンパイルされたものを見る。
4. 処理したファイルをプリントアウトする。

## 9.2 それでは使ってみましょう

ここでは cc2000 にインストールされている L<sup>A</sup>T<sub>E</sub>X (英語版、日本語版) を使う方法を紹介します。簡単な例を示しながら進めていきます。

まず、メニューのところから kterm を選択し、`mule first.tex &<return>` と入力し Mule を立ち上げ、L<sup>A</sup>T<sub>E</sub>X のソースファイルを作ります。(ここでは X 環境で Mule を利用する為にコマンドの最後に `&` を付けています。非 X 環境で Mule を利用する場合はこの `&` は必要ありません。)

<sup>1</sup> NeXT にある文机や Mac の EG-WORD、solo-writer や一太郎などで書いたものは使えません。

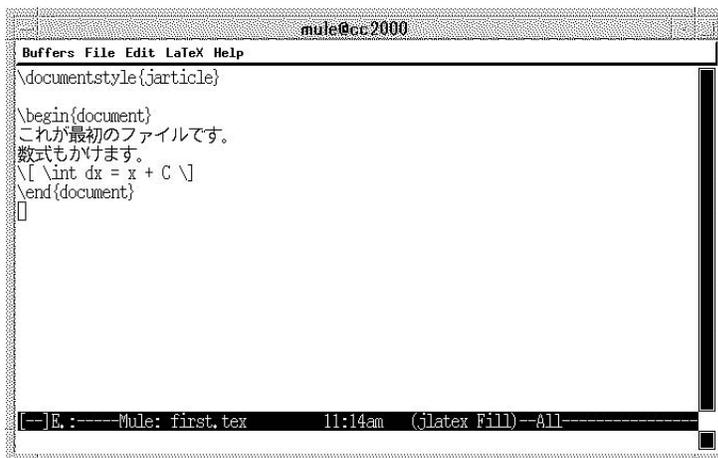


図 9.1 first.tex の画面

ファイルが出来たら Mule を C-x C-c で終了します。first.tex というファイルがあることを確認して、cc2000 のプロンプトで、jlatex first.tex<return>と入力します。そうすると次のように表示されます。

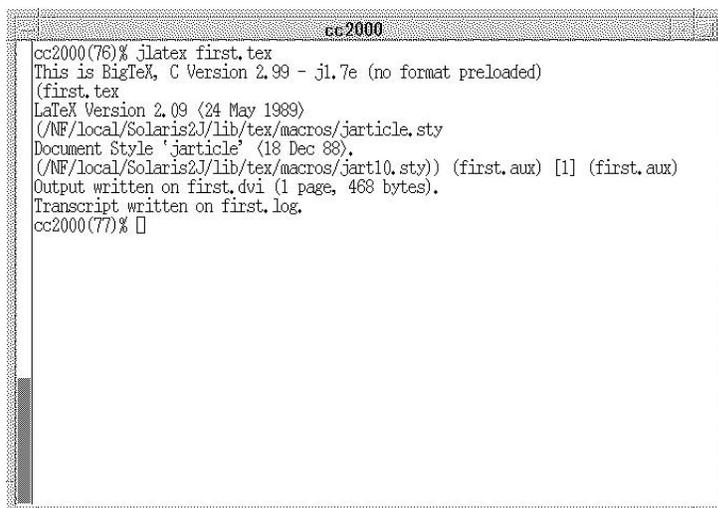


図 9.2 うまくいった時の画面

```
cc2000(82)% jlatex first.tex
This is BigTeX, C Version 2.99 - jl.7e (no format preloaded)
(first.tex
LaTeX Version 2.09 (24 May 1989)
(/NF/local/Solaris2J/lib/tex/macros/jarticle.sty
Document Style 'jarticle' (18 Dec 88).
(/NF/local/Solaris2J/lib/tex/macros/jart10.sty)) (first.aux)
! Missing $ inserted.
<inserted text>
      $
<to be read again> \intop
\int ->\intop \nolimits
L.6 [ \int
      dx = x + C \]
? x
No pages of output.
Transcript written on first.log.
cc2000(83)% □
```

図 9.3 エラーがあるときの画面

もしエラーがあった場合には、「？」マークが表示されますので、その後に  $x$  を入力するとコンパイルが中断されます。またどうしても処理が中断できない場合は `C-c` でプロセスを中断してください。その後、タイプミスの部分を探して直してからもう一度コンパイルの作業をして下さい。詳しいことは、第 9.14 節 (282 ページ) を参照して下さい。

さて、実際にコンパイルされたものがどのようなものに仕上がっているのか確認してみましょう。それには、プロンプトで `xdvi first.dvi &` と入力してみてください。そうすると、次のような画面が現れてきます。

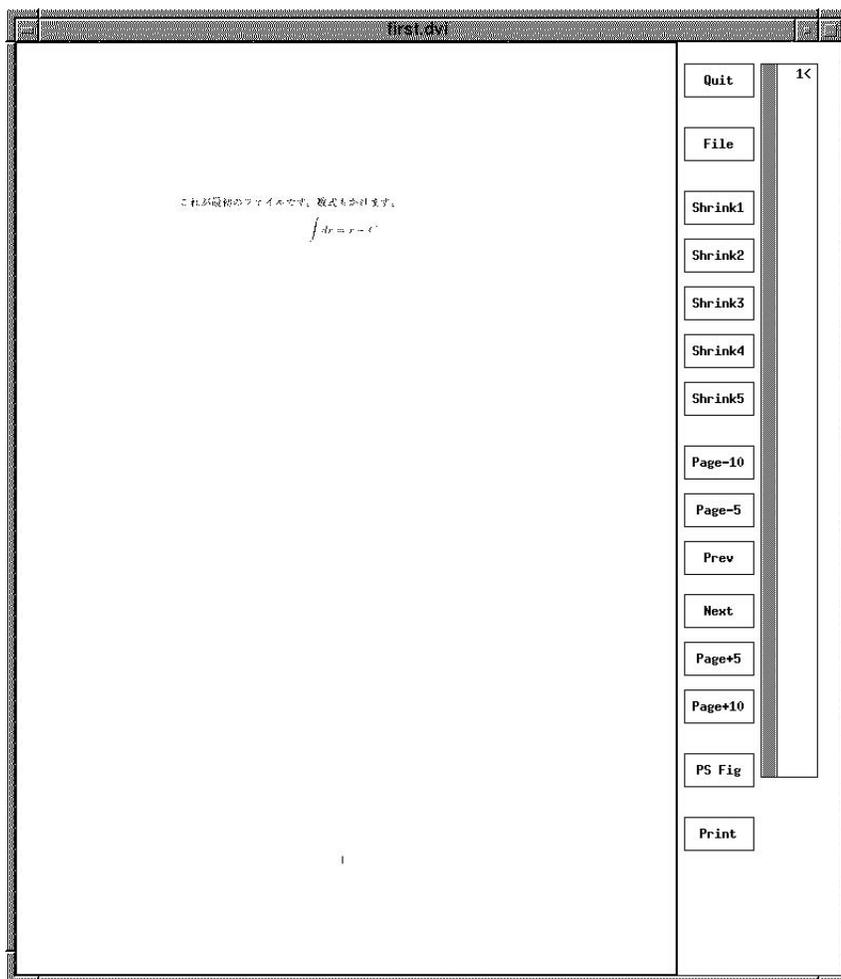


図 9.4 xdvi の画面

これが、コンパイルされて出来たものを表示した画面です。次にプリントアウトするのですが、このままではプリンターから出力できません。そのため、プリンターに出力できるようにして変換しておかなければなりません。

```
dvi2ps first.dvi > first.ps <return>
```

と入力します。この意味はプリンターが出力できるポストスクリプトという形式に変換する作業です。

```
cc2000
cc2000(79)% jlatex first.tex
This is BigTeX, C Version 2.99 - jl.7e (no format preloaded)
(first.tex
LaTeX Version 2.09 (24 May 1989)
(/NF/local/Solaris2J/lib/tex/macros/jarticle.sty
Document Style 'jarticle' (18 Dec 88).
(/NF/local/Solaris2J/lib/tex/macros/jart10.sty)) (first.aux) [1] (first.aux)
Output written on first.dvi (1 page, 468 bytes).
Transcript written on first.log.
cc2000(80)% jdvi2kps first.dvi > first.ps
[./NF/local/Solaris2J/lib/tex/jdvi2.ps] [1]
cc2000(81)% □
```

図 9.5 ポストスクリプトへの変換の画面

ようやくこれでプリントアウトできるようになりました。それでは、21 情報処理教室のプリンターから出力してみましょう。

```
lpr -Pcspr01 first.ps<return>
```

と入力してみてください。これも作業しているディレクトリの中で行なって下さい。21 情報処理教室のプリンター以外から出力する場合は cspr01 をそのプリンタ名に変更して下さい。

これで一通りの  $\text{\LaTeX}$  を使う作業の手順を紹介しました。また、dvi ファイルで出来具合を確認して、直接プリントアウトが出来ます。もう一度 `xdvi first.dvi &` と入力して下さい。

dvi ファイルが立ち上がりましたね。その画面の右下のところに、print ボタンというのがあります。このボタンを押すと、図 9.6 が現れます。



図 9.6 print ボタン

ここで、「print」と書いてあるところをドラッグして(マウスのボタンを押しながら)「All pages」と書かれているところでマウスをはなすと、すべてのページが cspr01 にプリントアウトされます。また、ある部分だけ印刷をしたいときは、第 9.15 節を参照して下さい。ここで注意しておかなければならないのは、

「プリント」のボタンを押した時、プリンターの名前が表示されますが、自分がいる情報処理教室のプリンターの名前になっているか確認して下さい。くれぐれもご注意を！

## 9.3 L<sup>A</sup>T<sub>E</sub>X におけるルール

### 例示の方法

これ以降では L<sup>A</sup>T<sub>E</sub>X の記述を例としてあげる場合に、以下のような記号を用いて表現します。

入力
この記号に続く記述は L <sup>A</sup> T <sub>E</sub> X のソースを表しています。
出力
この記号に続く内容が、上記の記述によって出力されます。

### 9.3.1 最低限のルール

文書ファイルの形式には次のような最低限のルールがあります。

- 文書ファイル名の最後に「.tex」をつけます (例: first.tex)。
- 文書ファイルの最初と最後に半角で次の様書きましょう。

```
\documentstyle{jarticle}

\begin{document}
:
:
:
\end{document}
```

`\begin{document}` から `\end{document}` までが文書の中身です。

- 使っている環境によって `\` (バックスラッシュ) が使えないときは半角の円記号 ¥ を使います。この 2 つは同じものだと思ってください。(文字コードが同じです。)

`\documentstyle{ ... }` と `\begin{document}` の間を「プリアンブル」といいます。このプリアンブルには紙の大きさや新しく定義するコマンド、環境などを書いておく場所です。これらについては後に詳しく述べていきます。言葉と役割を覚えて下さい。

### 9.3.2 ドキュメントスタイルについて

文書ファイルのはじめに書いた

```
\documentstyle{jarticle}
```

というような行は文書のスタイルを指定したものです。そしてこの中括弧 `{ }` の中は実際のスタイルの指定で次のようなものがあります。

`jarticle ..` 論文・雑誌の記事・短いレポートに使う最も一般的なスタイルです。

`jreport ...` 長いレポートなどに使うスタイルです。

jbook ..... 本などに使うスタイルです。見開きで見れるように奇数ページと偶数ページのレイアウトが違います。

これ以外にも幾つかのスタイルファイル がありますので、第 9.18 節を参照して下さい。

オプションについて

括弧 [ ] の中を「オプション」と呼びます。「オプション」は必要がなければ書く必要はありません。

```
\documentstyle[...]{jarticle}
```

以下にいくつかのオプションをあげておきます。

文字の大きさ .. 文書全体の文字の大きさを指定します<sup>2</sup>。10pt と 11pt と 12pt のいずれかの指定ができます。default では 10pt に指定されています。

紙の大きさ .... 出力する紙の大きさを指定します。a4(j)(A4)、a5(j)(A5)、b4(j)(B4)、b5(j)(B5) といった指定ができます。普通、日本語なら a4j を指定し、英語なら a4 を指定します。

紙の向き ..... 紙の向きを指定します。縦にして使うときは portrait、横にして使うときは landscape と指定します。しかし縦書きはできません。デフォルトでは portrait になっています。

その他 ..... 二段組にするときは jtwocolum と指定します。

上のようなオプションを複数指定してやりたいときは

```
\documentstyle[a4j,12pt]{jarticle}
```

のように半角のコンマ (,) で区切り、並べてやります。

### 9.3.3 見出しの種類

見出しの種類には次のようなものがあります。

種類	コマンド	jarticle	jreport, jbook
部	\part	Part I	Part I
章	\chapter	使えません	Chapter 1
節	\section	1	1.1
小節	\subsection	1.1	1.1.1
小小節	\subsubsection	1.1.1	1.1.1.1
段落	\paragraph	—	—
小段落	\subparagraph	—	—

スタイルファイルによって出力される形が違います。例として、jarticle の場合は

```
┌─── 入力 ───┐
│
│ \documentstyle[a4j]{jarticle}
│ \begin{document}
│ \part{第一部}
└──────────┘
```

<sup>2</sup> 注意：指定した大きさのフォントがないときはエラーがでます。

```
\section{第一節}
ここから始まります。
\subsection{第一小節}
ここから始まります。
\subsubsection{第一小小節}
ここから始まります。
\paragraph{第一段落}
ここから始まります。
\subparagraph{第一小段落}
\end{document}
```

---

出力

## Part I

# 第一部

## 1 第一節

ここから始まります。

### 1.1 第一小節

ここから始まります。

#### 1.1.1 第一小小節

ここから始まります。

第一段落   ここから始まります。

第一小段落   ここから始まります。

---

と出力されます。これ以外のスタイルでの形は御自分で試して下さい。

## 9.4 いろいろなコマンドと環境

### コマンドと環境の説明

このガイドで使われるコマンドはすべて「\」で始まるもので、タイプライタ体で書かれています。ここでいうタイプライタ体とは `typewriter style` のような書体です。一方、環境とは `\begin{...}\end{...}` の対で使うコマンドの一種みたいなものです。環境はすべて太字体で書かれています。よく現れてきますので、これから説明するコマンドと環境の使い方を是非覚えて下さい。

#### 9.4.1 特殊文字

L<sup>A</sup>T<sub>E</sub>X にはいくつかの「ルール」があります。このガイドでは、この「ルール」の説明と解説を一通りしていきます。本文でそのまま使える文字や記号は、アルファベットの小文字と大文字、数字の0~9と、! " ' ( ) = - ' @ + ; \* : , . / [ ] ? の19個の記号だけです。これ以外の記号で、# \$ % & ~ ^ \_ \ { } の10個の記号は本文中で単独に使用すると、「ヤバイ」こととなります<sup>3</sup>。というのも、これらの記号はそれぞれ意味を持っているからです。つまり、

- # は0~9までの数字と組み合わせて、parameter として使います。詳しくは第9.16節で述べます。
- \$ は、数字やアルファベットを\$ではさむと、そのはさんだ部分が数式モードになります。詳しくは参考図書をご覧ください。
- % は、%をつけた後の部分からその行の終わりまでを無視して出力します。つまり、自分がソースを書いている時の目印とか、その行のコメントとかに使います。

```
┌─── 入力 ───┐
What is done cannot be undone.%後悔先に立たず。
└─── 出力 ───┘
What is done cannot be undone.
```

このように、%以降の文章が出力されません。

- & は表を作る時の列を区切るのに使います。詳しいことは第9.10節で述べます。またこの記号は数式環境で複数行並べて表示させる時にも使います。この使い方は参考図書をご覧ください。
- ^ は上つき文字にします。ただし、数式の環境で使います。詳しくは参考図書をご覧ください。

```
┌─── 入力 ───┐
$ a^{11}+b^{11}=0 $
└─── 出力 ───┘
 $a^{11} + b^{11} = 0$ 
```

- \_ は下つき文字にします。ただし、これも数式モードで使います。詳しいことは参考図書をご覧ください。

```
┌─── 入力 ───┐
$ a_{11}+b_{11}=0 $
```

<sup>3</sup> 思った通りに出力してくれないばかりか、エラーになるかも知れない、ということです。

—— 出力 ——  
 $a_{11} + b_{11} = 0$

- ~ は一個分の空白を作ります。二個つければ 2 個分の空白を作ります。

—— 入力 ——  
 What is do~ne cannot be undone.  
 —— 出力 ——  
 What is do ne cannot be undone.

- \ はバックスラッシュとって、コマンドの始まりにつけます。前節で説明しました。
- { と } は、環境で用います。前節で説明しました。

このように、それぞれが特別な意味を持つため、これらを単独で使うことは避けた方が良いでしょう。どうしても使いたいのであれば、以下のようにして下さい。但し、大きさが少し大きめの文字として出力されます。

特殊文字	出力する為の記述	その結果表示される文字
#	\#	#
\$	\\$	\$
%	\%	%
&	\&	&
~	\~{ }	~
^	\^{ }	^
_	\_	_
\	\$\$backslash\$	\
{	\{	{
}	\}	}

˜、ˆ、\ については以下のようにして出力することも出来ます。

特殊文字	出力する為の記述	その結果表示される文字
˜	\symbol{"7E}	˜
ˆ	\symbol{"5E}	ˆ
\	{\tt\symbols{"5C}}	\

これらの記号について上記の方法が気に入らなければ、第 9.4.2 節で紹介する方法で出力して下さい。

また、使ってもエラーを起こしたりする事はないのですが < > | の三つの記号はそれぞれどういうわけか | ˆ — などという記号として出力されてしまいます<sup>4</sup>。< > | の記号を出力したい場合はやはり第 9.4.2 節で紹介する方法で出力して下さい。これらの記号については、第 9.4.2 節か、第 9.12 節で紹介する方法で出力して下さい。

<sup>4</sup> 但し TypeWriter フォントの時は大丈夫です。これが T<sub>E</sub>X の流儀だそうです。なぜこうなのか想像もつきませんが。

## 9.4.2 特殊文字でもそのまま出力する

アルファベットの小文字、大文字や数字、! “ ’ @ ‘ ( ) - = [ ] ; + : \* , . ? / の半角文字はそのまま出力されました。これら以外の半角文字は各種のコマンドや意味を持った特殊文字として  $\text{T}_{\text{E}}\text{X}$  に解釈されてしまい、そのままでは出力されませんでした。これらの記号を出力させるには `verbatim` という環境を使います。 `verbatim` 環境で書かれた文字はコマンドや特殊な意味を持つとは解釈されません。 `verbatim` 環境には二種類の使い方があります。以下にそれぞれ説明します。

記号一文字、もしくは一行におさまる文字列をそのまま出力する

`\verb` に続く記号でそのまま出力したい文字列をはさんでやります。どんな記号でも構いませんが、同じ記号ではさまないとエラーになりますので注意しましょう。以下に「 % (パーセント記号) 」を出力させる例を示します。

入力	<code>\verb!%!</code>
出力	<code>%</code>

入力	<code>\verb+sample サンプル ;:@]/., こんなぐあい+</code>
出力	<code>sample サンプル ;:@]/., こんなぐあい</code>

複数行にわたる文章をそのまま出力する

複数行にわたる文章をそのまま出力したければ、 `verbatim` 環境を使います。

入力
<code>\begin{verbatim}</code>
たとえば !@#\$\$^&*()<>;: なんかも
<code>[]{}'"\ -もみーんなこんなもんさ</code>
<code>\end{verbatim}</code>
出力
たとえば !@#\$\$^&*()<>;: なんかも
<code>[]{}'"\ -もみーんなこんなもんさ</code>

このようになります。

## 9.4.3 文字の空白

文字による空白には全角空白と半角空白があります。全角空白はそのままスペースキーで空けます。半角空白は何個空けても空白は1個分しか空けませんので注意しましょう。文字の空白は `\_` を使用します。`\_` は `space key` を一回押す、という意味です。そうすると、半角文字一個分だけ空白ができます。それ以上 `space key` をいれて空白を空けても1個分だけしか空白は空きません。半角空白を何個も出力するのは `\_` を繰り返しておきます。

入力	Time\shock
出力	Time shock
入力	Time\shock(space key を 6 回押しました。)
出力	Time shock
入力	Time\shock
出力	Time shock

#### 9.4.4 改行と改ページ

$\LaTeX$  で改行をするには一行分改行を入れるか、「`\`」というコマンドを用いる必要があります。一行分の改行を作ると  $\LaTeX$  のルールでは、改行されて少し頭が引っ込みます。つまり、段落ができるということです。一方、「`\`」は、強制改行のコマンドでただ改行するだけで頭は引っ込みません。

┌─── 入力 ───┐

改行を一行作ると改行されます。

そして、頭が引っ込みます。「`\verb!\\!`」はただ改行するだけです。`\\`  
 頭は引っ込みません。`\\`  
 このように改行されました。`\\`

─── 出力 ───

改行を一行作ると改行されます。

そして、頭が引っ込みます。「`\`」はただ改行するだけです。  
 頭は引っ込みません。  
 このように改行されました。

どうしても頭を引っ込めたくないならば、`\noindent` というコマンドがあります。逆に頭を引っ込めたいならば、`\indent` を用いればよいでしょう。

改ページをするには、`\newpage` という強制的に改ページするコマンドがあります。

#### 9.4.5 水平方向と垂直方向の空白

まず、水平方向の空白は`\hspace` があります。これは引数<sup>5</sup> と一緒に使います。

┌─── 入力 ───┐

水平方向に 0.5 cm 空白`\hspace{0.5cm}`を入れます。

─── 出力 ───

水平方向に 0.5 cm 空白 を入れます。

このように、「空白」と「を入れます」の間に 0.5 cm の空白ができました。

<sup>5</sup> 中括弧の中の数字です。

`\hspace` は右の方をプラスとすると、マイナスの方向つまり、左の方向にも空白を入れることができます。空白を入れるというよりも、空白を「取る」という感じです。

┌─── 入力 ───┐

水平方向に-0.5 cm 空白`\hspace{-0.5cm}`を入れます。

─── 出力 ───┐

水平方向に-0.5 cm 空白

このように、出力されます。よく見ますと、文字が重なってしまいました。これは「を入れます。」という言葉が前に 0.5cm 移動したと考えて下さい。文章中に `\hspace` を入れたらそこから水平方向にずれましたが、改行した後で、`\hspace` を使っても水平方向にはずれません。そこで、`\hspace*{...}` というコマンドを使います。例えば

┌─── 入力 ───┐

```
あいうえお\\
\hspace{2cm}あいうえお\\
\hspace*{2cm}あいうえお\\
あいうえおあいうえおあいうえお\\
```

─── 出力 ───┐

```
あいうえお
      あいうえお
      あいうえお
あいうえおあいうえおあいうえお
```

このようにすれば、行の始めから空白を入れることができます。  
次に垂直方向の空白です。コマンドは `\vspace` です。使い方は `\hspace` と同じ使い方です。

┌─── 入力 ───┐

垂直方向に 3.5 cm 空白`\vspace{3.5cm}`を入れます。\\  
こうなります。

─── 出力 ───┐

垂直方向に 3.5 cm 空白を入れます。

┌─── 出力 ───┐  
こうなります。

このように、3.5 cm 垂直方向に空白ができました。  
`\vspace` も同様に、マイナスの方向に空白を入れることができます。

┌─── 入力 ───┐

垂直方向に-0.5 cm 空白\space{-0.5cm}を入れます。\\  
こんな感じになります。

┌─── 出力 ───┐

垂直な感じになります。

御覧のように文字が重なってしまいました。これも「こんな感じになります」という文が上に 0.5 cm 移動したことになります。

L<sup>A</sup>T<sub>E</sub>X で使える長さの単位を以下に挙げておきます。

単位の名前	長さ	幅
cm	センチメートル	+ +
em	大文字の M の横幅	+ +
ex	小文字の m の横幅	+ +
in	インチ (約 25.4mm)	+ +
pc	パイカ (12pt, 約 4.21mm)	+ +
pt	ポイント (0.35mm)	++
mm	ミリメートル	++

出力に出ているのはその単位で出力される間隔です。<sup>6</sup>

全角文字の横の大きさは `zw` を単位としています。これはスタイルファイルのところを `j` の付いたもの<sup>7</sup> に限り使えます。

### 9.4.6 引用

引用を行うものとして、`quote` と `quotation` の二つの環境があります。`quote` 環境は次のように短い文章を引用するのに向いています。

┌─── 入力 ───┐

```
\begin{quote}
我々は京都産業大学の学生である。
\end{quote}
```

┌─── 出力 ───┐

我々は京都産業大学の学生である。

一方、`quotation`<sup>8</sup> 環境は比較的長い文章や、複数のパラグラフがあるような文章を引用するのに向いています。

┌─── 入力 ───┐

<sup>6</sup> この原稿は縮小されて印刷されていますので実際はもう少し広いです。

<sup>7</sup> つまり、`jarticle`、`jreport`、`jbook` の場合です。

<sup>8</sup> 出力例を見ていただいたらわかるように左右に余白ができています。

```
\begin{quotation}
```

7月初旬のおそろしく暑い時分のこと、とある夕方近く、一人の若い男が、C-横町の借家人からまた借りていた自分の部屋から街路に出て、なんとなく心のきまらないさまで、のろのろとK-橋の方へ歩いて行った。

```
\end{quotation}
```

—— 出力 ——

7月初旬のおそろしく暑い時分のこと、とある夕方近く、一人の若い男が、C-横町の借家人からまた借りていた自分の部屋から街路に出て、なんとなく心のきまらないさまで、のろのろとK-橋の方へ歩いて行った。

#### 9.4.7 箇条書き

箇条書きをするのには、以下の3つの方法があります。

- `itemize` は各々の箇条の先頭に・をつけた (記号つき箇条書き) 形です。
- `enumerate` は各々の箇条の先頭に番号をつけた形 (番号つき箇条書き) です。
- `description` は各々の箇条の先頭にラベルをつけた形 (見出しつき箇条書き) です。

箇条書きをするにはこれらを環境として使い、それぞれの行の先頭に`\item`をつけます。

##### 記号つき箇条書き

記号つき箇条書きとは箇条の先頭に記号をつけた形で出力します。

—— 入力 ——

```
\begin{itemize}
```

```
\item {\bf itemize}は各々の箇条の先頭に・をつけた (記号つき箇条書き) 形です。
```

```
\item {\bf enumerate}は各々の箇条の先頭に番号をつけた形 (番号つき箇条書き) です。
```

```
\item {\bf description}は各々の箇条の先頭にラベルをつけた形 (見出しつき箇条書き) です。
```

```
\end{itemize}
```

—— 出力 ——

- `itemize` は各々の箇条の先頭に・をつけた (記号つき箇条書き) 形です。
- `enumerate` は各々の箇条の先頭に番号をつけた形 (番号つき箇条書き) です。
- `description` は各々の箇条の先頭にラベルをつけた形 (見出しつき箇条書き) です。

のようになります。

## 番号つき箇条書き

これは箇条の先頭に番号をつけた形になります。

┌─── 入力 ───┐

```
\begin{enumerate}
\item {\bf itemize}は各々の箇条の先頭に・をつけた（記号つき箇条書き）形です。
\item {\bf enumerate}は各々の箇条の先頭に番号をつけた形（番号つき箇条書き）です。
\item {\bf description}は各々の箇条の先頭にラベルをつけた形（見出しつき箇条書き）です。
\end{enumerate}
```

─── 出力 ───

1. **itemize** は各々の箇条の先頭に・をつけた（記号つき箇条書き）形です。
2. **enumerate** は各々の箇条の先頭に番号をつけた形（番号つき箇条書き）です。
3. **description** は各々の箇条の先頭にラベルをつけた形（見出しつき箇条書き）です。

となります。

また、`\item`の後に、`[1a.]`などの番号を入れておきますと

┌─── 入力 ───┐

```
\begin{enumerate}
\item[1a.] {\bf itemize}は各々の箇条の先頭に・をつけた（記号つき箇条書き）形です。
\item[1b.] {\bf enumerate}は各々の箇条の先頭に番号をつけた形（番号つき箇条書き）です。
\item[1c.] {\bf description}は各々の箇条の先頭にラベルをつけた形（見出しつき箇条書き）です。
\end{enumerate}
```

─── 出力 ───

- 1a. **itemize** は各々の箇条の先頭に・をつけた（記号つき箇条書き）形です。
- 1b. **enumerate** は各々の箇条の先頭に番号をつけた形（番号つき箇条書き）です。
- 1c. **description** は各々の箇条の先頭にラベルをつけた形（見出しつき箇条書き）です。

このようになります。

## 見出しつき箇条書き

この環境は前の二つと少し違って以下のようにします。

まず、`\item`の後に「見出し」をつけます。この見出しが前の二つでいう「記号」や「数字」に当たります。例で見てみましょう。

┌─── 入力 ───┐

```

\begin{description}
\item[自然数] 自然数とは、
\item[整数] 整数とは、
\item[有理数] 有理数とは、
\item[無理数] 無理数とは、
\item[複素数] 複素数とは、
\end{description}

```

—— 出力 ——

自然数 自然数とは、  
整数 整数とは、  
有理数 有理数とは、  
無理数 無理数とは、  
複素数 複素数とは、

これらを入れ子状にしても使えます。例えば、`itemize` 環境では

┌—— 入力 ——┐

```

\begin{itemize}
\item 箇条書き
  \begin{itemize}
\item 箇条書き
  \begin{itemize}
\item 箇条書き
  \begin{itemize}
\item 箇条書き
\end{itemize}
\end{itemize}
\end{itemize}
\end{itemize}
\end{itemize}

```

—— 出力 ——

- 箇条書き
  - 箇条書き
    - \* 箇条書き
      - 箇条書き

となります。`enumerate` 環境では

┌—— 入力 ——┐

## 1. 箇条書き

### (a) 箇条書き

#### i. 箇条書き

#### A. 箇条書き

---

となります。このような形で入れ子状の箇条書きができます<sup>9</sup>。

しかし、この入れ子状も3回までが限度です。それ以上入れるとエラーになります。注意しましょう。

---

<sup>9</sup> 箇条書きの記号や番号は変えることができます。例えば、数字の番号ではなく、アルファベットの番号にすることもできますが、ここでは説明しません。

#### 9.4.8 右寄せ、中央寄せ、左寄せ

文章を右に寄せたり、中央に寄せたり、左に寄せたりするには、`flushright`, `center`, `flushleft` を環境として使います。

入力	出力
<pre>\begin{flushright} 文章を右によせます。 \end{flushright} \begin{center} 文章を中央に寄せます。 \end{center} \begin{flushleft} 文章を左に寄せます。 \end{flushleft}</pre>	<pre>文章を右によせます。  文章を中央に寄せます。  文章を左に寄せます。</pre>

という具合になります。

右寄せ、中央寄せ、左寄せは他に `\raggedleft`, `\raggedright`, `\centering` でも行えます。但しこれを書いた以降の全ての文に影響を与えますので、どちらかという文書全体の形を決める宣言として考えて下さい。どのような形に整形されるかは、既に出来上がっている複数行を含む文書で試して下さい。

#### 9.4.9 文字の大きさ

基本となる文字のサイズは 10 pt です。その他に 11 pt, 12 pt が指定できました。これ以外で文字の大きさを文中で指定することができます。以下にあげたように 10 個の文字の大きさがあります。`\tiny` はルビサイズです。`\footnotesize` は脚注の文字の大きさです。`\normalsize` は普通の文字の大きさです。`\large` からは文字の大きさが少しずつ大きくなっていきます。(注意)<sup>10</sup> ここでひとこと。文中ではあまり文字のサイズを変えることはおすすめできません<sup>11</sup>。

入力	出力	入力	出力
<code>\tiny</code>	<code>tiny</code>	<code>\large</code>	<code>large</code>
<code>\scriptsize</code>	<code>scriptsize</code>	<code>\Large</code>	<code>Large</code>
<code>\footnotesize</code>	<code>footnotesize</code>	<code>\LARGE</code>	<code>LARGE</code>
<code>\small</code>	<code>small</code>	<code>\huge</code>	<code>huge</code>
<code>\normalsize</code>	<code>normalsize</code>	<code>\Huge</code>	<code>Huge</code>

<sup>10</sup> ここで、印刷されているのは縮小して印刷していますので、若干、文字が小さくなっています。

<sup>11</sup> けっこう、読みにくいものです。注意しましょう。

## 9.4.10 書体

入力	欧文出力	日本語出力
<code>\rm</code>	roman	普通の欧文書体
<code>\bf</code>	<b>boldface</b>	太字体
<code>\it</code>	<i>italic</i>	斜体
<code>\sl</code>	<i>slanted</i>	傾斜体
<code>\sf</code>	sans serif	サンセリフ体
<code>\sc</code>	SMALL CAPS	大文字体
<code>\tt</code>	Typewriter	タイプライタ体
<code>\em</code>	<i>emphasize</i>	強調体

このように、日本語は太字体にしか変わりません。また、強調体は斜体と同じ文字となります。

これらの命令はひとたび使いますと、以降の文章をすべてその書体にしてしまいますので、それを防ぐにはその書体にしたい範囲を{ ... }で囲っておきますとその括弧の中だけがコマンドの文体になります。

入力
abcdefg 京都産業大学\\
{\rm abcdefg 京都産業大学}\\
{\tt abcdefg 京都産業大学}\\
{\bf abcdefg 京都産業大学}\\
{abc{\bf de}f 京都{\bf 産業}大学}

出力
abcdefg 京都産業大学
abcdefg 京都産業大学
abcdefg 京都産業大学
<b>abcdefg 京都産業大学</b>
abcdef 京都産業大学

文字の大きさと種類を同時に指定する時は書体を先に指定し、その次に大きさを指定します。

	入力	出力
書体が先で大きさが後	<code>{\huge {\bf UNIX}}</code>	<b>UNIX</b>
大きさが先で書体が後	<code>{\bf {\huge UNIX}}</code>	UNIX

下の例では太文字体の指定が無効になってしまいました。

さらに、`{\huge \bf UNIX}`のように中の括弧を取り除くことができます。

入力	出力
<code>{\huge \bf UNIX}</code>	<b>UNIX</b>

ここで、注意しておかなくてはならないことは、文字の大きさと書体を同時に指定した時、大きさが変わるの漢字と roman 体、bold 体のみである、ということです。つまり、文字を大きくする命令は漢字と roman 体、bold 体にしか働かないということです。従って、これ以外の書体で大きくしたいときには、参考図書をご覧ください。

## 9.5 表題

### 9.5.1 タイトルの作り方

タイトルをつけるには、以下にあげるコマンドを用います。

- `\title{タイトル名}`—ここにタイトル名を書きます。
- `\author{著者名}`—著者が複数の時は `\and` コマンドを使って連名にします。
- `\date{日付}`—`\date` が省略された時はその原稿ファイルが  $\text{\LaTeX}$  に向け、結果ファイルが作られた日になります。ここまでが一組になります。これらのうち一つでも欠けるとエラーになります。
- `\maketitle`—タイトルを出力します。

これらは `\begin{document}` の後ならどこでも構いません。

そして、`\maketitle` は `\title{タイトル名}` などの後におきます。後の例を参考にして下さい。

### 9.5.2 概要の作り方

タイトルの次は、概要の作り方です。これは論文などではよく見かけられるものです。やり方は、

```
\begin{abstract}
  概要の内容
\end{abstract}
```

とします。

┌─── 入力 ───┐

```
\title{ここにタイトルがきます。}
\author{ここが書いた人の名前\and 共同者の名前}
\date{出来上がった日付}
\maketitle
\begin{abstract}
  今までワープロを使っていると、数学の式で、複雑なものはできませんでした。ところが、 $\text{\LaTeX}$  は、それをきれいな形で出力できます。また、論文や、レポートなどにも手軽に使えます。ただ、ルールがあって、ワープロのようにすぐ使えるわけではありません.....。
\end{abstract}
```

─── 出力 ───

ここにタイトルがきます。

ここが書いた人の名前                  共同者の名前

出来上がった日付

Abstract

今までワープロを使っていると、数学の式で、複雑なものはできませんでした。ところが、 $\text{\LaTeX}$  は、それをきれいな形で出力できます。また、論文や、レポートなどにも手軽に使えます。ただ、ルールがあって、ワープロのようにすぐ使えるわけではありません。....

ここで注意してほしいのは、ドキュメントスタイルを指定するとそれに合わせて Part や Chapter が変わるのと同じように、表題もスタイルファイルによって変わります。つまり、上の例は `jarticle` で書かれたものです。 `jreport` と `jbook` では一枚の紙に、タイトル、著者、日付を出力します。そして、ページをかえて、Abstract を出力し、ページをかえて、本文が始まります。ところが、 `jbook` では Abstract が使えません。

ドキュメントスタイル	title	author	date	Abstract
<code>jarticle</code>				
<code>jreport</code>				
<code>jbook</code>				×

## 9.6 傍注

傍注として本文の横に「余白だ！」と出力するには、 `\marginpar {余白だ!}` というコマンドを使いま 余白だ！  
 す。傍注が本文の右に来るか、左に来るかは `\documentstyle{...}` で指定した本文のスタイルに依存しま  
 す。どこに出るかはいろいろ試して見るのがいいでしょう。

`\marginpar [左余白]{右余白}` と書けば、右に余白があった場合にはそこに「右余白」、左に余白があつた場合にはそこに「左余白」と出力されます。

## 9.7 脚注

脚注<sup>12</sup> を出力するには、 `\footnote` というコマンドを使います。 `\footnote{ここに出る注釈を脚注と呼びます。}` という具合です。本文中の `\footnote` が現れたところのフットノートナンバーが振られ、括弧の部分がページの下に書かれます。

## 9.8 相互参照

論文やレポートといった文章を書いている時、「第 2.5.7 節を参照して下さい」とか、「(6) 式に (7) を代入して...」といったことがよくあります。これをそのまま本文に書いていると、文章を加筆したり削除することで番号が違ってくる時があります。そうすると、このようなことが起きるたびにいちいち番号をかえていかななくてはなりません。そこで、この番号振りを自動でやってくれるのが、 `\label{...}` というコマンド<sup>13</sup> です。例えば

```
\section{...}\label{bun}
```

としておきますと、 `{...}` というセクションに「bun」というラベルがついたことになります。このラベルには何でも構いませんが、 `\{, \}, ...` といった特殊な記号は駄目です。ラベルを出力するには相棒となる `\ref{...}` を使います。また、ページ番号を出力したい時には `\pageref{bun}` とすると、出力されます。

### 入力例

```
\subsection{相互参照}\label{bun}
... 第\ref{bun}節参照...
... \pageref{bun}...
```

### 出力例

<sup>12</sup> ここに出る注釈を脚注と呼びます。

<sup>13</sup> ラベルは本文中ならオールマイティにつけることができます。

... 第 9.14.2 節参照...

...284 ページ参照...

### 注意

`\label` コマンドを使った時は、必ずコンパイルを 2 回以上行って下さい。というのは、一連のコンパイルの手順は、以下のような形になります。

`.tex` ファイル

(jlatex コンパイル)↓    ↑ (error 訂正)

`.aux` ファイル

`.log` ファイル

`.dvi` ファイル

`.tex` ファイルを作って、一回でもコンパイルをすると、この残り 3 つのファイル<sup>14</sup> が出来上がります。

例えば、`'S'` というラベルを二重に使うと

—— 出力 ——

```
LaTeX Warning: Label 'S' multiply defined.
```

と表示され、`\ref{S}` のところには、「??」という記号が付けられます。

- `.tex` ファイル 原稿のファイルです。
- `.aux` ファイル は、補助ファイルです。ここに、相互参照のラベルのついたものが書かれます。
- `.log` ファイル は、エラーメッセージや実行状態が書かれます。ここにエラーの原因が書いてありますので、もしコンパイルのときエラーを起こして直したけれども、それでもまだエラーを起こすような時、原因がよくわからなければここを見ると解決することがあります。
- `.dvi` ファイル 出力結果のファイルです。

相互参照に利用されるファイルは上記の `.aux` ファイルです。これは 1 回だけではただラベルが列挙されただけで `\label` と `\ref` の対応関係がついていません。例えばあるファイルをコンパイルして

—— 出力 ——

```
LaTeX Warning: Label(s) may have changed. Rerun to get cross-references right.
```

という行が表示された時はもう一度コンパイルの作業をすることによって、きちんと `\label` と `\ref` の対応関係がつかます。つまり、相互参照を含むドキュメントを修正した場合に、参照している章節の番号やページ位置が保証されるのは内容を修正せずに二回連続でコンパイルした時だけだということです。

<sup>14</sup> これ以外にも L<sup>A</sup>T<sub>E</sub>X コンパイルした時に作られるファイルはあります。

## 9.9 箱

L<sup>A</sup>T<sub>E</sub>X ではいろいろな箱を作ることができます。枠付きの箱、点線の箱、枠なしの箱があります。

### 9.9.1 一行に収まる文字列を囲む

枠付きの箱 `fbox` と `framebox`

枠付きの箱を作るコマンドには

- `\fbox`
- `\framebox`

があります。

`\fbox` は一行分の箱を作ります。使い方は

```
—— 入力 ——  
\fbox{ここに囲みたい文を入れます。}  
—— 出力 ——  
ここに囲みたい文を入れます。
```

となります。この`\fbox` は一行分の箱しかできません。つまり、2行以上にわたる文もすべて一行にしてしまいます。例えば

```
—— 入力 ——  
\fbox{あいうえおかきくけこさしすせそたちつてと\ なにぬねのはひふへほまみむめもやゆよりるれるわを}  
—— 出力 ——  
あいうえおかきくけこさしすせそたちつてとなにぬねのはひふへほまみむめもやゆよりるれるわを
```

のように、一行の箱になります<sup>15</sup>。`\fbox` はキチキチの箱でしたが、`\framebox` というコマンドを使えばこの枠を自由に変えることができます。

`\framebox[長さ][位置]{囲みたい文}`とします。文の位置は指定しないと中央になります。位置の指定には他に

- `[r]` 右に合わせます。
- `[l]` 左に合わせます。

があります。いろいろな長さの箱ができますが、これも一行分の箱しかできません。例えば

```
—— 入力 ——  
\framebox[10cm][l]{あいうえおかきくけこさしすせそ}  
—— 出力 ——  
あいうえおかきくけこさしすせそ
```

となります。

<sup>15</sup> 一行に収まらない文は、はみ出してでも一行の箱にします。

もっとでかい箱を作るには `\framebox` を  
`\framebox(横の大きさ, 縦の大きさ) [位置] {箱の中に入る文}`  
のように使います。縦、横の大きさの単位は pt(ポイント) を使います。位置は、中央が基本で

- [t] 上の中央に合わせます。
- [b] 下の中央に合わせます。
- [r] 右の真中に合わせます。
- [l] 左の真中に合わせます。
- [tr] 右上の角に合わせます。
- [tl] 左上の角に合わせます。
- [br] 右下の角に合わせます。
- [bl] 左下の角に合わせます。

があります。例えば

```
—— 入力 ——  
\framebox(300,30) [bl] {あいうえおかきくけこさしすせそ}  
—— 出力 ——  
あいうえおかきくけこさしすせそ
```

となります。これも、一行分しか入りません。試してみてください。

#### 点線の箱 `dashbox`

点線の箱を作るには、`\dashbox` というコマンドを使います。使い方は  
`\dashbox{点線の長さ}(横の大きさ, 縦の大きさ) [位置] {箱の中に入る文}`  
のようにします。例えば

```
—— 入力 ——  
\dashbox{1}(300,30) [bl] {あいうえおかきくけこさしすせそ}  
—— 出力 ——  
あいうえおかきくけこさしすせそ
```

となります。

#### 枠なしの箱 `makebox`

今までは枠がありましたが、`\makebox` というコマンドを使えば枠のない箱を作ることできます。使い方は `\framebox` と同じです。

`\makebox(横の大きさ, 縦の大きさ) [位置] {箱の中に入る文}`

例えば

```
┌─── 入力 ───┐
\makebox(300,30)[t1]{あいうえおかきくけこさしすせそ}
└─── 出力 ───┘
                あいうえおかきくけこさしすせそ
```

となります。これも、一行分しか入りません。

## 9.9.2 複数行にわたる文の箱を作る

今までは一行分の箱しかできませんでしたが、複数行の箱を作るには`\parbox`というコマンドと、`minipage`環境があります。

`parbox` を使った箱

`\parbox` を使うには、以下のようにします。  
`\parbox[基準線]{横幅の長さ}{文章}`  
とします。とにかく例を見て下さい。 はわかりやすくするために`\parbox` と`\parbox` を区切っています<sup>16</sup>。

```
┌─── 入力 ───┐
\parbox[t]{4.5cm}{あいうえおかきくけこさしすせそ}
\parbox[t]{4.5cm}{あいうえおかきくけこさしすせそたちつてとなにぬねのはひふへほ}
\parbox[t]{4.5cm}{あいうえおかきくけこさしすせそたちつてとなにぬねのはひふへほまみむめもやゆ
よりりるれろわゐゑを}
└─── 出力 ───┘
あいうえおかきくけこさしす   あいうえおかきくけこさしす   あいうえおかきくけこさしす
せそ                             せそたちつてとなにぬねのは   せそたちつてとなにぬねのは
                                 ひふへほ                             ひふへほまみむめもやゆよ
                                                                           りるれろわゐゑを
```

違う例として

```
┌─── 入力 ───┐
\parbox[b]{4.5cm}{あいうえおかきくけこさしすせそ}
\parbox[b]{4.5cm}{あいうえおかきくけこさしすせそたちつてとなにぬねのはひふへほ}
\parbox[b]{4.5cm}{あいうえおかきくけこさしすせそたちつてとなにぬねのはひふへほまみむめもやゆ
よりりるれろわゐゑを}
```

<sup>16</sup> この は別のものでもなくとも構いません。

出力

		あいうえおかきくけこさしす
	あいうえおかきくけこさしす	せそたちつてとなにぬねのは
あいうえおかきくけこさしす	せそたちつてとなにぬねのは	ひふへほまみむめもやゆよら
せそ	ひふへほ	りるれるるわゐゑを

となります。例で見ていただいたように、基準線とは `\parbox` を上で揃えるか `[t]` 下で揃えるか `[b]`、ということです。枠の大きさを決めてしまえば後はその大きさに沿って文章を改行していきます<sup>17</sup>。これは枠がありませんでしたが、枠をつけるには `\fbox` と一緒に使います。つまり

入力

```
\fbox{\parbox[t]{4.5cm}{あいうえおかきくけこさしすせそ}}
\fbox{\parbox[t]{4.5cm}{あいうえおかきくけこさしすせそたちつてとなにぬねのはひふへほ}}
\fbox{\parbox[t]{4.5cm}{あいうえおかきくけこさしすせそたちつてとなにぬねのはひふへほまみむめもやゆよらりるれるるわゐゑを}}
```

出力

あいうえおかきくけこさしす せそ	あいうえおかきくけこさしす せそたちつてとなにぬねのは ひふへほ	あいうえおかきくけこさしす せそたちつてとなにぬねのは ひふへほまみむめもやゆよら りるれるるわゐゑを
---------------------	--	--

と枠ができ上がります。

### 9.9.3 minipage 環境

これは `\parbox` を発展させたものです。以下のように使います。

```
\begin{minipage}[基準線]{横幅の長さ}
.....
\end{minipage}
```

入力

```
\begin{minipage}[t]{8cm}
{\bf minipage}環境は、小さなページのようになります。そして、
脚注もできます\footnote{footnote もつけることができます。}。\\
\end{minipage}
```

出力

minipage 環境は、小さなページのようになります。  
そして、脚注もできます<sup>a</sup>。

<sup>a</sup> footnote もつけることができます。

<sup>17</sup> 改行するというより、文章をつめていく、と言った方がいいかも知れません。

ここで、[t] は基準線を決めるものでした。そして、8 cm の小さなページができあがりました。さらに、このページを枠で囲ってみましょう。

入力

```
\fbox{
  \begin{minipage}[t]{8cm}
  {\bf minipage}環境は小さなページのようになります。そして、
  脚注もできます\footnote{footnote もつけることができます。}。
  \end{minipage}
}
```

出力

minipage 環境は小さなページのようになります。そして、脚注もできます<sup>a</sup>。

<sup>a</sup> footnote もつけることができます。

次の例は minipage 環境をたくさん使ったものです。

入力

```
\fbox{
  \begin{minipage}{15cm}
  minipage 環境です。\\
  \fbox{
    \begin{minipage}{10cm}
    minipage 環境です。\\
    \fbox{
      \begin{minipage}{5cm}
      minipage 環境です。\\
      \fbox{
        \begin{minipage}{4cm}
        minipage 環境です。\\
        \end{minipage}}
      \end{minipage}}
    \end{minipage}}
  \end{minipage}}
\end{minipage}}
```

出力

minipage 環境です。  
minipage 環境です。  
minipage 環境です。  
minipage 環境です。

このような使い方も出来ます。

## 9.10 表

表を作るのが `tabular`<sup>18</sup> 環境です。表を作る環境は `tabular` 環境の他に、`tabbing` 環境がありますが、ここでは省略させていただきます。

```
\begin{tabular}{列指定}
表全体
\end{tabular}
```

このような形で作ります。列指定とは表の中にある要素の配置を指定します。列指定には以下のものがあります。

- l... 右寄せ (left)
- c... 中央 (center)
- r... 左寄せ (right)

そして、列の数だけこれを並べます。表の要素一個一個を区切るには「&」を入れます。そして、列の最後には、「\\」をつけます。くれぐれも列指定の数と表の要素の列の数を同じにして下さい。

```
┌─── 入力 ───┐
\begin{tabular}{lcr|lcr}
品名 & 値段 & 数量 & 品名 & 値段 & 数量 \\
莓 & 200 円 & 2 ケース & 葡萄 & 400 円 & 3 ケース \\
西瓜 & 1000 円 & 2 玉 & 檸檬 & 100 円 & 10 個 \\
\end{tabular}
└─── 出力 ───┘


| 品名 | 値段     | 数量    | 品名 | 値段    | 数量    |
|----|--------|-------|----|-------|-------|
| 莓  | 200 円  | 2 ケース | 葡萄 | 400 円 | 3 ケース |
| 西瓜 | 1000 円 | 2 玉   | 檸檬 | 100 円 | 10 個  |


```

これに罫線を入れてみましょう。縦の罫線は列指定のところに入れてたいところに「|」を入れます。横の罫線は、`\hline` です。「\\」の後に置きます。「||」や`\hline\hline` とすれば2重線になります。

```
┌─── 入力 ───┐
\begin{tabular}{|l|c|r||l|c|r|} \hline
品名 & 値段 & 数量 & 品名 & 値段 & 数量 \\ \hline
莓 & 200 円 & 2 ケース & 葡萄 & 400 円 & 3 ケース \\ \hline
西瓜 & 1000 円 & 2 玉 & 檸檬 & 100 円 & 10 個 \\ \hline
\end{tabular}
└─── 出力 ───┘


| 品名 | 値段     | 数量    | 品名 | 値段    | 数量    |
|----|--------|-------|----|-------|-------|
| 莓  | 200 円  | 2 ケース | 葡萄 | 400 円 | 3 ケース |
| 西瓜 | 1000 円 | 2 玉   | 檸檬 | 100 円 | 10 個  |


```

<sup>18</sup> 「タビュラー」と読んで下さい。

となります。もう少し複雑な表を作ってみましょう。

#### 入力

```
\begin{tabular}{|c|c|c|}\hline時刻 & \multicolumn{2}{c|}{平日} \\ \cline{2-3} & 神社発 & 本学発 \\ \hline 10 & 5,10,20 & 0,10,15 \\ & 25,30,45 & 20,35,45 \\ \hline\end{tabular}
```

#### 出力

時刻	平日	
	神社発	本学発
10	5,10,20	0,10,15
	25,30,45	20,35,45

ここで使われている`\multicolumn`というコマンドは表のある部分だけ、例えばこの表であれば「平日」という項が「神社発」と「本学発」の欄にまたがっています。つまり、複数の項目にわたって出力したり、その欄だけ他と違うような形で出力したい時に使います。使い方は、

```
\multicolumn{またがる項の数}{列指定}{項目の内容}
```

とします。ここでは、`\multicolumn{2}{c|}{平日}`としました。また、`\cline{2-3}`というコマンドは、2項目と3項目にだけ線を引きます。だから、ここの数字をかえれば部分的に線を引くことができます。

これ以上の詳しい説明はやめにしておきますが、付録の参考文献にあげておきました`LATEX`に関する本を参考して、もっと複雑な表を作ってみて下さい。

## 9.11 絵

文章を書いていると、絵や図が必要になることがあります。そんな時役に立つが figure 環境です。ここでは、絵や図の作り方は説明しませんが、xpaint とか tgif などで作ったとします。ただし、これらの絵や図をセーブする時は必ず PostScript 形式で行なって下さい。さて、注意しなければならないことがあります。それは \documentstyle のところで、「[epsbox]」というオプションをつけなくてはなりません。これがないとエラーになります。

では、実際にやってみましょう。mathematica で描いたトーラス曲線を EPS ファイルで保存します。

┌── 入力 ──┐

```
\documentstyle{jarticle}
\begin{document}
\begin{figure}[hbt]
  \begin{center}
    \leavevmode
    \psbox[scale=0.5]{tolus-curve.ps}
  \end{center}
  \label{tolus}
  \caption{トーラス曲線}
\end{figure}
\end{document}
```

── 出力 ──

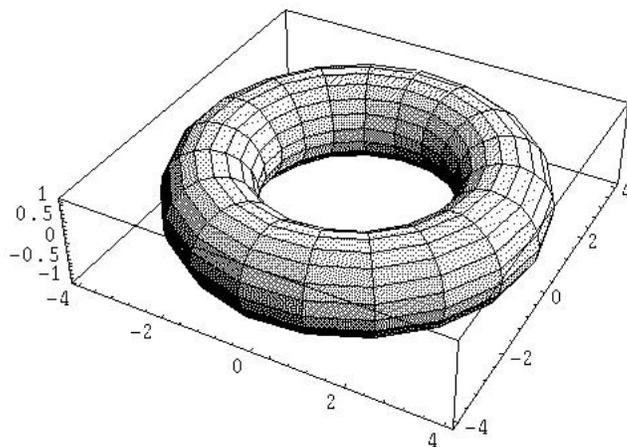


図 9.7 トーラス曲線

それでは、説明していきます。

- まず、figure 環境を書きます。これは単に絵や図をはめ込むための領域を確保するためのものです。
- オプションとして、[hbt] を指定しておきます

- `center` 環境は、単に真中に図を置きたかったから入れました<sup>19</sup>。
- これが重要なコマンド `\psbox` です。このコマンドが絵や図を文章の中にはめてくれます。
- `[scale=0.5]` は、図の倍率の指定です。ここでは、元の大きさの 0.5 倍に指定しています。
- `{tolus-curves.ps}` は、ファイル名です。「.ps」は省略しても構いません。
- `\label` コマンドは、第 9.14.2 節を参照して下さい。
- `\caption` コマンドは、絵や図の下に、表題をつけてくれます。別にいらなければ書かなくても結構です。

これらをとにかく呪文のように打って下さい。これで、図を文章の中にはめ込むことが出来ます。

#### コメント

オプションとして `[hbtpt]` と指定しましたが、これらは次のような意味です。

- h** here の頭文字で、「ここにはめ込む」という感じです。
- b** 「そのページの下にはめ込む」という感じです。入らなければ次のページの下にはめ込みます。
- t** 「そのページの上にはめ込む」という感じです。こちらも入らなければ次のページの上にはめ込みます。
- p** 独立したページにはめ込みます。つまり、その絵を出力し改ページをします。

それぞれの意味を書きましたが、 $\text{\LaTeX}$  はオプションの順番通りに作業してみて一番いい状態のものを出力してくれます。しかし、おそらくあなたの希望通りには絵をはめ込んでくれないでしょう。つまり、「ここに絵を入れたい」といっても  $\text{\LaTeX}$  はやってくれるかどうかわかりません。そこで、`\newpage` コマンドを使って何とかしてみるしかありません<sup>20</sup>。

<sup>19</sup> 別にあってもなくても構いません。

<sup>20</sup> 実際、このガイドもいくつかの `\newpage` を用いて見易いようにしています。

このように PostScript 形式のファイルであれば、 $\text{\LaTeX}$  にはめ込むことができます。もう一つ、`tgif` で書いた `coordinates.obj` という図<sup>21</sup> もはめ込むこともできます。

入力

```
\begin{figure}[htbp]
  \begin{center}
    \input{coordinates.tps}\label{fig-1}
  \end{center}
  \caption{球面座標の直交基底}
\end{figure}
```

出力

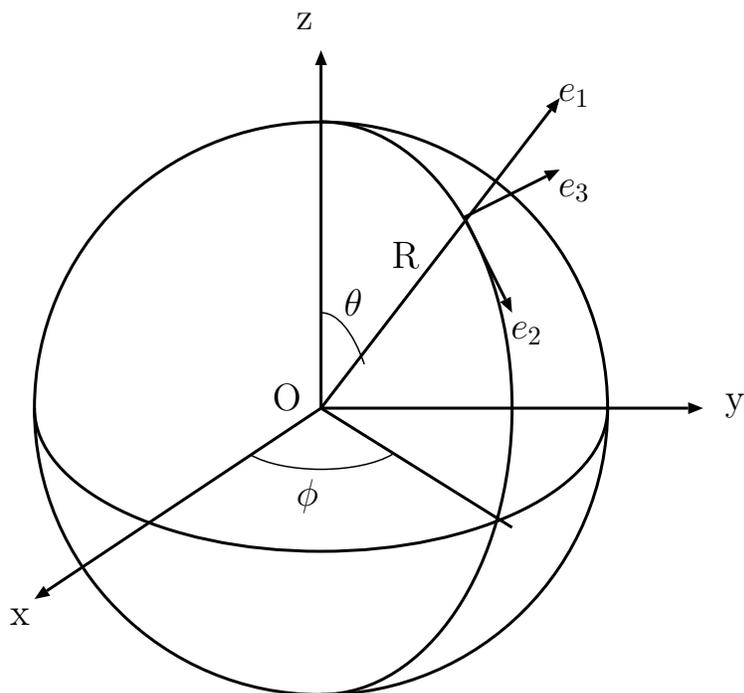


図 9.8 球面座標の直交基底

<sup>21</sup> `tgif` のところを参照して下さい。

## 9.12 L<sup>A</sup>T<sub>E</sub>X で扱える記号

L<sup>A</sup>T<sub>E</sub>X で扱える各種の記号類を挙げます。

### 9.12.1 雑記号

入力	出力	入力	出力	入力	出力	入力	出力
<code>\S</code>	§	<code>\P</code>	¶		—		
<code>\lq</code>	‘	<code>\rq</code>	’	<code>\lbrack</code>	[	<code>\rbrack</code>	]
<code>\slash</code>	/	<code>\copyright</code>	©	<code>\aa</code>	å	<code>\AA</code>	Å
<code>\dag</code>	†	<code>\ddag</code>	‡	<code>\TeX</code>	T <sub>E</sub> X	<code>\LaTeX</code>	L <sup>A</sup> T <sub>E</sub> X
<code>\dots</code>	...	-	-	--	-	---	—

独立した記号ではありませんが、望みの文字に下線を引くことも出来ます。

入力 `\underbar{a}`

出力 a

入力 ここらで一発 `\underbar{下線}` が欲しいなあ

出力 ここらで一発 下線 が欲しいなあ

### 9.12.2 空白を空ける文字

ここに挙げるコマンドはそれぞれ一定の大きさの空白を作ります。幅ならば右方向に、高さならば下方向という具合です。但し大きさが負の場合、幅ならば左方向に、高さならば上方向に空白が出来ることになります。長さの単位については 第 9.4.5 節を参照して下さい。

入力	意味	入力	意味	入力	意味
<code>\space</code>	空白一文字	<code>\empty</code>	何もなし	<code>\null</code>	幅ゼロの箱
<code>\thinspace</code>	1/6em の幅	<code>\negthinspace</code>	-1/6em の幅	<code>\enspace</code>	1/2em の幅
<code>\enskip</code>	1/2em の幅	<code>\quad</code>	1em の幅	<code>\qqquad</code>	2em の幅
<code>\,</code>	3mu の幅	<code>\$\&gt;\$</code>	4mu の幅	<code>\$\;\$</code>	5mu の幅
<code>\!</code>	-3mu の幅				
<code>\smallskip</code>	3pt の高さ	<code>\medskip</code>	6pt の高さ	<code>\bigskip</code>	12pt の高さ

`$\>$`と`$\;$`については\$ で囲まれていることから判るように数式中で使うべきものですが、入れる場所がなかったのでここに採録しました。本来は数式中で微妙な隙間を作る為に、もともとあった`\,`に追加されたような気がします。なお、これらが使っている mu という長さの単位についてはその定義が判りませんでした。1mu で 1 ミリあるかないかの短い長さです。実際に試してみてください。

### 9.12.3 アクセントなど

入力	出力	入力	出力	入力	出力	入力	出力
<code>\`{a}</code>	à	<code>\' {a}</code>	á	<code>\v{a}</code>	ă	<code>\u{a}</code>	ǎ
<code>\={a}</code>	ā	<code>\^ {a}</code>	â	<code>\. {a}</code>	à	<code>\H{a}</code>	ǎ
<code>\~{a}</code>	ã	<code>\" {a}</code>	ä	<code>\d{a}</code>	ą	<code>\c{c}</code>	ç
<code>\b{aa}</code>	aa	<code>\t{aa}</code>	ââ				

例)

入力 Poincar\’e

出力 Poincaré

\”を除いては{}は不要です。つまり、\’\_a\_で à になります。また、ç は c 専用の記号です。

#### 9.12.4 ヨーロッパ系言語特有の記号

入力	出力	入力	出力	入力	出力	入力	出力
<	ı	>	ı	\l	ł	\L	L
\ss	ß	\oe	œ	\ae	æ	\o	ø
\AE	Æ	\OE	Œ	\O	Ø		

## 9.13 数式環境で使える記号

L<sup>A</sup>T<sub>E</sub>X の数式で扱える各種の記号類を挙げます。

### 9.13.1 雑記号

入力	出力	入力	出力	入力	出力	入力	出力
<code>\hbar</code>	$\hbar$	<code>\mho</code>	$\Omega$	<code>\surd</code>	$\surd$	<code>\angle</code>	$\angle$
<code>\neq, \ne</code>	$\neq$	<code>\colon</code>	$:$	<code>\bowtie</code>	$\bowtie$	<code>\amalg</code>	$\amalg$
<code>\lhook</code>	$\hookleftarrow$	<code>\rhook</code>	$\hookrightarrow$	<code>\ldotp</code>	$\cdot$	<code>\cdotp</code>	$\cdot$
<code>\ldots</code>	$\dots$	<code>\cdots</code>	$\cdots$	<code>\vdots</code>	$\vdots$	<code>\ddots</code>	$\ddots$
<code>\Vert, \parallel, \lvert</code>	$\parallel$	<code>\models</code>	$\models$	<code>\uplus</code>	$\uplus$	<code>\cdot</code>	$\cdot$
<code>\backslash, \setminus</code>	$\setminus$	<code>\vert, \mid</code>	$ $	<code>\doteq</code>	$\doteq$	<code>\equiv</code>	$\equiv$
<code>\approx</code>	$\approx$	<code>\sim</code>	$\sim$	<code>\simeq</code>	$\simeq$	<code>\asymp</code>	$\asymp$
<code>\rangle</code>	$\rangle$	<code>\langle</code>	$\langle$	<code>\rbrace</code>	$\rbrace$	<code>\lbrace</code>	$\lbrace$
<code>\rceil</code>	$\rceil$	<code>\lceil</code>	$\lceil$	<code>\rfloor</code>	$\rfloor$	<code>\lfloor</code>	$\lfloor$
<code>\aleph</code>	$\aleph$	<code>\imath</code>	$\imath$	<code>\jmath</code>	$\jmath$	<code>\ell</code>	$\ell$
<code>\wp</code>	$\wp$	<code>\Re</code>	$\Re$	<code>\Im</code>	$\Im$	<code>\partial</code>	$\partial$
<code>\infty</code>	$\infty$	<code>\prime</code>	$'$	<code>\emptyset</code>	$\emptyset$	<code>\nabla</code>	$\nabla$
<code>\top</code>	$\top$	<code>\bot</code>	$\bot$	<code>\perp</code>	$\perp$	<code>\neg</code>	$\neg$
<code>\forall</code>	$\forall$	<code>\exists</code>	$\exists$	<code>\flat</code>	$\flat$	<code>\natural</code>	$\natural$
<code>\sharp</code>	$\sharp$	<code>\dagger</code>	$\dagger$	<code>\ddagger</code>	$\ddagger$	<code>\smallint</code>	$\int$
<code>\clubsuit</code>	$\clubsuit$	<code>\diamondsuit</code>	$\diamondsuit$	<code>\heartsuit</code>	$\heartsuit$	<code>\spadesuit</code>	$\spadesuit$
<code>\triangleleft</code>	$\triangleleft$	<code>\diamond</code>	$\diamond$	<code>\Diamond</code>	$\Diamond$	<code>\bullet</code>	$\bullet$
<code>\triangleright</code>	$\triangleright$	<code>\wedge</code>	$\wedge$	<code>\vee</code>	$\vee$	<code>\wr</code>	$\wr$
<code>\bigtriangleup</code>	$\bigtriangleup$	<code>\div</code>	$\div$	<code>\times</code>	$\times$	<code>\mp</code>	$\mp$
<code>\bigtriangledown</code>	$\bigtriangledown$	<code>\pm</code>	$\pm$	<code>\smile</code>	$\smile$	<code>\frown</code>	$\frown$
<code>\cap</code>	$\cap$	<code>\cup</code>	$\cup$	<code>\sqcap</code>	$\sqcap$	<code>\sqcup</code>	$\sqcup$
<code>\sqsubset</code>	$\sqsubset$	<code>\sqsupseteq</code>	$\sqsupseteq$	<code>\sqsubset</code>	$\sqsubset$	<code>\sqsupset</code>	$\sqsupset$
<code>\odot</code>	$\odot$	<code>\oslash</code>	$\oslash$	<code>\otimes</code>	$\otimes$	<code>\ominus</code>	$\ominus$
<code>\oplus</code>	$\oplus$	<code>\circ</code>	$\circ$	<code>\bigcirc</code>	$\bigcirc$	<code>\triangle</code>	$\triangle$
<code>\propto</code>	$\propto$	<code>\ast</code>	$\ast$	<code>\star</code>	$\star$	<code>\Box</code>	$\Box$
<code>\dashv</code>	$\dashv$	<code>\vdash</code>	$\vdash$	<code>\leq</code>	$\leq$	<code>\geq</code>	$\geq$
<code>\succ</code>	$\succ$	<code>\prec</code>	$\prec$	<code>\succeq</code>	$\succeq$	<code>\preceq</code>	$\preceq$
<code>\supset</code>	$\supset$	<code>\subset</code>	$\subset$	<code>\supseteq</code>	$\supseteq$	<code>\subseteq</code>	$\subseteq$
<code>\in</code>	$\in$	<code>\ni</code>	$\ni$	<code>\gg</code>	$\gg$	<code>\ll</code>	$\ll$

先に上げた記号の中で、同じ記号を出力しますが、空白領域が微妙に違うものがあります。例えば、`\cdotp` と `\cdot` は

入力	<code>\$GG\cdotp GG\cdot GGG\$</code>
出力	$GG \cdot GG \cdot GGG$

また、`\bot` と `\perp` も微妙に違います。

入力	<code>\$FF\perp FF\bot FF\$</code>
出力	$FF \perp FF \bot FF$

独立した記号ではありませんが、`\not` を使って否定型を作ることできます。

入力	<code>a\not\in X</code>
出力	$a \notin X$
入力	<code>A\not\supset B</code>
出力	$A \not\supset B$

少し大きめの記号です。

入力	出力	入力	出力	入力	出力	入力	出力
<code>\coprod</code>	$\coprod$	<code>\intop</code>	$\intop$	<code>\prod</code>	$\prod$	<code>\sum</code>	$\sum$
<code>\bigvee</code>	$\bigvee$	<code>\bigwedge</code>	$\bigwedge$	<code>\bigcap</code>	$\bigcap$	<code>\bigcup</code>	$\bigcup$
<code>\bigotimes</code>	$\bigotimes$	<code>\bigoplus</code>	$\bigoplus$	<code>\bigodot</code>	$\bigodot$	<code>\ointop</code>	$\ointop$
<code>\bigsqcup</code>	$\bigsqcup$						

矢印です。

入力	出力	入力	出力
<code>\Longrightarrow</code>	$\Longrightarrow$	<code>\Longleftarrow</code>	$\Longleftarrow$
<code>\Rightarrow</code>	$\Rightarrow$	<code>\Leftarrow</code>	$\Leftarrow$
<code>\longrightarrow</code>	$\longrightarrow$	<code>\longleftarrow</code>	$\longleftarrow$
<code>\rightarrow , \to</code>	$\rightarrow$	<code>\leftarrow , \gets</code>	$\leftarrow$
<code>\longmapsto</code>	$\longmapsto$	<code>\mapsto</code>	$\mapsto$
<code>\Leftrightarrow</code>	$\Leftrightarrow$	<code>\longleftarrowrightarrow</code>	$\longleftrightarrow$
<code>\iff</code>	$\iff$	<code>\leadsto</code>	$\leadsto$
<code>\Leftrightarrow</code>	$\Leftrightarrow$	<code>\leftrightarrow</code>	$\leftrightarrow$
<code>\hookrightarrow</code>	$\hookrightarrow$	<code>\hookleftarrow</code>	$\hookleftarrow$
<code>\rightharpoonup</code>	$\rightharpoonup$	<code>\leftharpoonup</code>	$\leftharpoonup$
<code>\rightharpoondown</code>	$\rightharpoondown$	<code>\leftharpoondown</code>	$\leftharpoondown$
<code>\rightleftharpoons</code>	$\rightleftharpoons$		
<code>\Uparrow</code>	$\Uparrow$	<code>\Downarrow</code>	$\Downarrow$
<code>\uparrow</code>	$\uparrow$	<code>\downarrow</code>	$\downarrow$
<code>\Updownarrow</code>	$\Updownarrow$	<code>\updownarrow</code>	$\updownarrow$
<code>\nearrow</code>	$\nearrow$	<code>\searrow</code>	$\searrow$
<code>\nwarrow</code>	$\nwarrow$	<code>\swarrow</code>	$\swarrow$

続いて単独の記号ではなく、文字の上に付ける記号です。付ける元の文字をコマンドの後ろに`{ }`で囲んで与えます。

入力	出力	入力	出力	入力	出力	入力	出力
<code>\acute{a}</code>	á	<code>\grave{a}</code>	à	<code>\ddot{a}</code>	ä	<code>\tilde{a}</code>	ã
<code>\bar{a}</code>	ā	<code>\breve{a}</code>	ă	<code>\check {a}</code>	ǎ	<code>\hat{a}</code>	â
<code>\vec{a}</code>	→	<code>\dot{a}</code>	·				

再び単独の記号ではなく、文字の上に付ける記号です。今度は付ける元の文字の長さに合わせて記号の長さも変化します。同じく付ける元の文字をコマンドの後ろに{ }で囲んで与えます。

入力	出力	入力	出力
<code>\underline{abc}</code>	<u>abc</u>	<code>\overline{abc}</code>	$\overline{abc}$
<code>\widetilde{abc}</code>	$\widetilde{abc}$	<code>\wodehat{abc}</code>	$\widehat{abc}$
<code>\overrightarrow{abc}</code>	$\overrightarrow{abc}$	<code>\overleftarrow{abc}</code>	$\overleftarrow{abc}$
<code>\overbrace{abc}</code>	$\overbrace{abc}$	<code>\underbrace{abc}</code>	$\underbrace{abc}$
<code>\sqrt{abc}</code>	$\sqrt{abc}$	<code>\sqrt[3]{abc}</code>	$\sqrt[3]{abc}$

何と云えば良いのでしょうか？とにかく見て下さい。

入力	出力
<code>\{abc\}choose{def}</code>	$\binom{abc}{def}$
<code>\{abc\}brack{def}</code>	$\left[ \begin{matrix} abc \\ def \end{matrix} \right]$
<code>\{abc\}brace{def}</code>	$\left\{ \begin{matrix} abc \\ def \end{matrix} \right\}$

### 9.13.2 ギリシャ文字

ギリシャ文字の一覧表です。<sup>22</sup>

ギリシャ文字				
文字の名前	大文字		小文字	
	出力	入力	出力	入力
alpha	A		$\alpha$	<code>\alpha</code>
beta	B		$\beta$	<code>\beta</code>
gamma	$\Gamma$	<code>\Gamma</code>	$\gamma$	<code>\gamma</code>
delta	$\Delta$	<code>\Delta</code>	$\delta$	<code>\delta</code>
epsilon	E		$\epsilon, \varepsilon$	<code>\epsilon, \varepsilon</code>
zeta	Z		$\zeta$	<code>\zeta</code>
eta	H		$\eta$	<code>\eta</code>
theta	$\Theta$	<code>\Theta</code>	$\theta, \vartheta$	<code>\theta, \vartheta</code>
iota	I		$\iota$	<code>\iota</code>
kappa	K		$\kappa$	<code>\kappa</code>
lambda	$\Lambda$	<code>\Lambda</code>	$\lambda$	<code>\lambda</code>
mu	M		$\mu$	<code>\mu</code>
nu	N		$\nu$	<code>\nu</code>
xi	$\Xi$	<code>\Xi</code>	$\xi$	<code>\xi</code>
omicron	O		$o$	$o$ (英語の $o$ と同じ)
pi	$\Pi$	<code>\Pi</code>	$\pi, \varpi$	<code>\pi, \varpi</code>
rho	P		$\rho, \varrho$	<code>\rho, \varrho</code>
sigma	$\Sigma$	<code>\Sigma</code>	$\sigma, \varsigma$	<code>\sigma, \varsigma</code>
tau	T		$\tau$	<code>\tau</code>
upsilon	$\Upsilon$	<code>\Upsilon</code>	$\upsilon$	<code>\upsilon</code>
phi	$\Phi$	<code>\Phi</code>	$\phi, \varphi$	<code>\phi, \varphi</code>
chi	X	<code>\Chi</code>	$\chi$	<code>\chi</code>
psi	$\Psi$	<code>\Psi</code>	$\psi$	<code>\psi</code>
omega	$\Omega$	<code>\Omega</code>	$\omega$	<code>\omega</code>

<sup>22</sup> 大文字で抜けているのは英語の大文字と同じだからです。

### 9.13.3 関数

入力	出力	入力	出力	入力	出力	入力	出力
<code>\log</code>	log	<code>\lg</code>	lg	<code>\ln</code>	ln		
<code>\lim</code>	lim	<code>\limsup</code>	lim sup	<code>\liminf</code>	lim inf		
<code>\sin</code>	sin	<code>\arcsin</code>	arcsin	<code>\sinh</code>	sinh		
<code>\cos</code>	cos	<code>\arccos</code>	arccos	<code>\cosh</code>	cosh		
<code>\tan</code>	tan	<code>\arctan</code>	arctan	<code>\tanh</code>	tanh		
<code>\sec</code>	sec	<code>\csc</code>	csc	<code>\max</code>	max	<code>\min</code>	min
<code>\sup</code>	sup	<code>\inf</code>	inf	<code>\arg</code>	arg	<code>\ker</code>	ker
<code>\dim</code>	dim	<code>\hom</code>	hom	<code>\det</code>	det	<code>\exp</code>	exp
<code>\Pr</code>	Pr	<code>\gcd</code>	gcd	<code>\deg</code>	deg	<code>\bmod</code>	mod

最後に挙げた mod に関してはもう一つ以下のような形が用意されています。

入力	<code>\pmod{n}</code>
出力	$(\text{mod } n)$

## 9.14 エラーの対処

### 9.14.1 エラーの表示

サンプルファイルとして、`abc.tex` を作ります。

```
┌─── 入力 ───┐
\documentstyle{jarticle}
\begin{document}
\[ \left\{ \begin{array}{c}
x'=ax+by\\
y'=cx+dy
\end{array} \right. \]
\end{document}
└──────────┘
```

このファイルを `jlatex` でコンパイルしますと

```
─── 出力 ───
cc2000(83)% jlatex abc.tex
This is BigTeX, C Version 2.99 - j1.7e (no format preloaded)
(abc.tex
LaTeX Version 2.09 <24 May 1989>
(/NF/local/Solaris2J/lib/tex/macros/jarticle.sty
Document Style 'jarticle' <18 Dec 88>.
(/NF/local/Solaris2J/lib/tex/macros/jart10.sty)) (abc.aux) [1] (abc.aux)
Output written on abc.dvi (1 page, 456 bytes).
Transcript written on abc.log.
└──────────┘
```

と表示された時は正しくコンパイルされたこととなります。正しい例を出しましたが、どこか間違っている例を出しましょう。

**ここからはまず読んで、それから実際にやってみてください。**

```
┌─── 入力 ───┐
\documentstyle{jarticle}
\begin{document}
\[ \left\{ \begin{array}{c}
x'=ax+by\\
y'=cx+dy
\end{array} \right \]
\end{document}
└──────────┘
```

さきほどの `abc.tex` のところで、6行目の `\right.` の「`.`」をはずして見ます。そして、コンパイルしますと

—— 出力 ——

```
cc2000(84)% jlatex abc.tex
This is BigTeX, C Version 2.99 - j1.7e (no format preloaded)
(abc.tex
LaTeX Version 2.09 <24 May 1989>
(/NF/local/Solaris2J/lib/tex/macros/jarticle.sty
Document Style 'jarticle' <18 Dec 88>.
(/NF/local/Solaris2J/lib/tex/macros/jart10.sty)) (abc.aux)
No file abc.aux.
! Missing delimiter (. inserted).
<to be read again>
                \edef
\@latexerr #1#2->\edef
                \@tempc {#2}\expandafter \errhelp \expandafter {\@tem...

\]->\relax \ifmmode \ifinner \@badmath
                \else $$\fi \else \@badmath \fi \igno...

1.6 \end{array} \right \]

?
```

と、プロンプトが入力待ちの状態で止まります。これが間違った時に表示されるエラー表示です。それでは、重要な部分だけを解説していきましょう。

—— 出力 ——

```
cc2000(83)% jlatex abc.tex
This is BigTeX, C Version 2.99 - j1.7e (no format preloaded)
(abc.tex
LaTeX Version 2.09 <24 May 1989>
(/NF/local/Solaris2J/lib/tex/macros/jarticle.sty
Document Style 'jarticle' <18 Dec 88>.
(/NF/local/Solaris2J/lib/tex/macros/jart10.sty)) (abc.aux)
No file abc.aux.
```

1 行目から 8 行目までは、エラーとは関係ありませんので飛ばします。

—— 出力 ——

```
! Missing delimiter (. inserted).
```

ここからが、エラーを表示しているところです。これは区切り文字が見つからないという事です。つまり、対になるべきところがないという事です。次に

—— 出力 ——

```
\@latexerr #1#2->\edef
```

これはエラーの種類を表しています。つまり、`latexerr` とあるように、コマンドの使い方が間違っている、という事です。そして、その場所が

—— 出力 ——

```
1.6 \end{array} \right \]
```

6行目のところにあります、という意味です。最後の行に「?」が表示されて、コンパイルの作業が中断していますので、「x」と打ってみてください。そうすると

—— 出力 ——

```
? x
```

```
No pages of output.
```

```
Transcript written on abc.log.
```

と表示されてるはずですが。この後は、`Mule` を立ちあげて、間違っているところを直していきましょう。

### 9.14.2 ちょっと違うやり方

先ほどは「?」の後に、「x」を打ちましたが、それ以外にも次のようなものがあります。

`s`, または `r` これはエラーがあっても構わず最後までコンパイルをします。従って、いっぱいエラーがあると思われるファイルは使用しない方がいいでしょう。

`h` これは英語で何が間違っているかを教えてください。

`e` このキーを打つと、自動的に `mule` が立ち上がり、かつエラーを起こしている行の先頭にカーソルが出るようになっています。

みなさん、それぞれ試してみてください。

### 9.14.3 エラーの種類

ここではいくつかのエラーの表示を見てください。

- ! Missing \$ inserted.
  1. 数式環境でエラーをしています。
- ! Undefined control sequence.
  1. 定義されていないコマンドを使った。
  2. コマンドの後に空白がない。
  3. 余計なところで「\」がついてしまったためにコマンドとしてみなしてしまっている。
- ! \begin{eqnarray} ended by \end{document}.
  1. \begin、\end の対応関係がおかしい。つまり、どちらかが多かったり、ぬけている。
- ! Missing } inserted.
- ! Missing { inserted.
  1. 括弧の対応が正しくありません。
- Runaway argument?
  1. \section において、閉じ括弧を忘れている。
  2. \verb ではさんでいる記号が違う。など、コマンドの使い方が間違っている。
- ! Extra alignment tab has been changed to \cr.
  1. 表を宣言した時に指定した列の数よりソースにある列の数の方が多い。
  2. 列の最後に改行マーク「\」が抜けている。
- ! Environment ?????? undefined.
  1. 環境の命令の綴りが違っている。
  2. 定義していない環境を用いている。

などなど、エラーはたくさんありますので列挙するのはやめます。とにかく、エラーを起こしている行番号を頼りに直して下さい。自分の勘違いやタイプミスがほとんどです。あきらめずに一つずつ直していきましょう。

もし、エラーの対処についてわからなければ、 $\text{\LaTeX}$  の達人に作っている自分のファイルとそのファイル名の拡張子が .log のファイルの両方を見てもらいましょう。

## 9.15 部分印刷する方法

さて、完成した文章に少し手を加えて書き直した時など、最初から印刷するのは時間と紙がもったいないと思います。そこで、部分印刷するには `dvi2ps` コマンドを次のように使います。

```
cc2000(3)% dvi2ps -f 開始ページ -t 終了ページ foo.dvi > foo.ps
```

としますと、開始ページから終了ページまでの PostScript ファイル `foo.ps` が出来上がります。

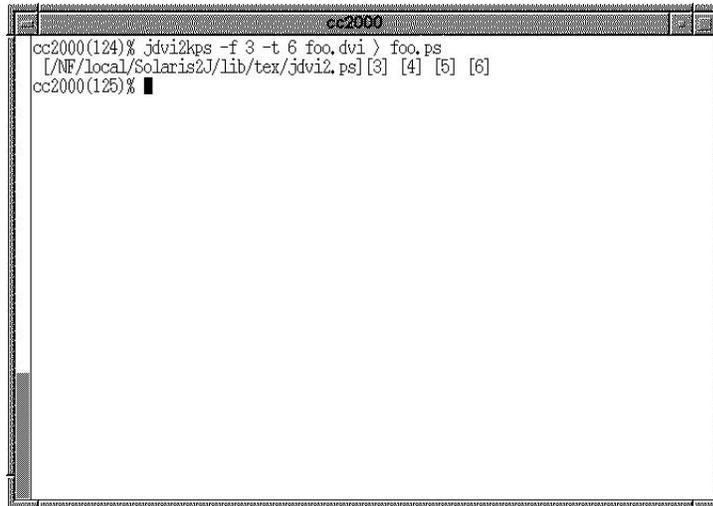


図 9.9 dvi ファイルから部分だけ取り出す方法

ここで、一旦 PostScript ファイルを確認してみましょう。cc2000 のプロンプトで `ghostview foo.ps` と入力して下さい。そうすると

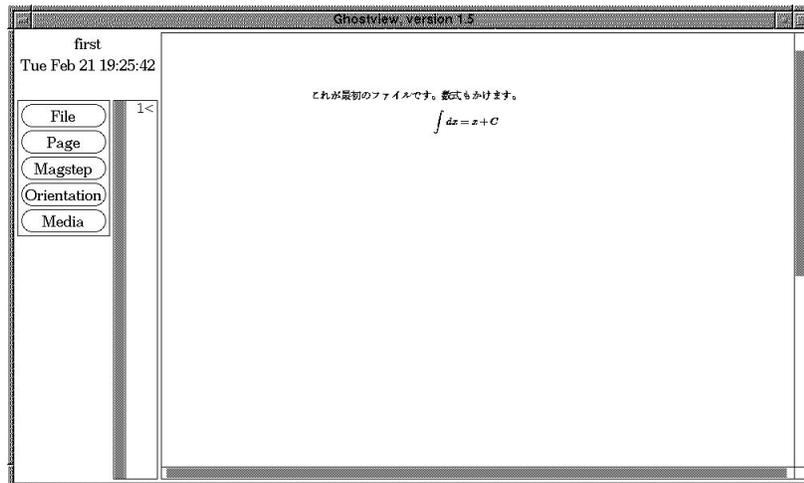


図 9.10 PostScript 形式のファイルの確認

このような画面が現れてきます。これで、PostScript 形式のファイルを見ることが出来ます。これで良ければプロンプトで

```
lpr -Pcspr01 foo.ps
```

とすれば、プリントアウトされます。

また、xdvi の画面でプリントボタンをクリックすると、次のような画面が現れて



図 9.11 プリントボタン

**Print** と表示されたところをドラッグして、「Current page」のところでもうすボタンを離すと、そのページだけを印刷することができます。

部分印刷に関することではないのですが、一枚の紙 (例えば、A4 サイズ) に縮小して何ページものを詰め込む方法があります。cc2000 に nup というアプリケーションをインストールしています。これを用います。そして、次のようにして使ってください。

```
┌─── 入力 ───┐  
cc2000(1)% dvi2ps foo.dvi | nup -2 > foo.ps  
└──────────┘
```

と入力すれば、A4 の紙に 2 ページ分が縮小されます。nup のオプションは

- 2 用紙一枚に 2 枚分縮小して印刷する。
- 4 用紙一枚に 4 枚分縮小して印刷する。
- 8 用紙一枚に 8 枚分縮小して印刷する。
- 1 縮小して印刷されたページを枠で囲む。
- n 各ページを縮小しないで印刷する。これは B5 でフォーマットした出力を B4 用紙に縮小しないで並べて印刷します。
- p 出力する用紙サイズを指定する。

があります。縮小しますので、字が小さくなって読めなくなる場合があります。注意しましょう。

## 9.16 自分の命令 (マクロ)

L<sup>A</sup>T<sub>E</sub>X では自分だけの命令 (マクロ) を作ることができます。これは既存の命令を組み合わせで定義し作ります。

### 9.16.1 簡単な命令を作ってみよう

いま用紙の中央に次のように書こうとします。

京都産業大学

これを実現するためには

—— 入力 ——

```
\begin{center}
京都産業大学
\end{center}
```

とします。

これと同じ働きをする `\KSU` という命令<sup>23</sup> を作ります。そのためには文書ファイルにこの命令を使うより以前のところでつぎのように `\KSU` を定義しておきます。

```
\newcommand{\KSU}{\begin{center}京都産業大学\end{center}}
```

ファイルのこれ以降の場所で `\KSU` と書けば

```
\begin{center}京都産業大学\end{center}
```

と書いたのと同じ働きをします。

### 9.16.2 引数を持っている命令の作り方

さっきの `\KSU` を「京都産業大学」をセンタリングするだけでなく、ほかの単語もセンタリングできるようにバージョンアップしてみましょう。

次のようにします。

```
\newcommand{\namae}[1]{\begin{center} #1 \end{center}}
```

このようにすればこれ以降、`\namae{...}` と書けば (... のところに何か語を書くんですよ) 真ん中に単語がレイアウトされます<sup>24</sup>。ここで `[1]` は引数の数、`#1` は引数の入る場所です。引数とは、そのマクロに対応する文字列のことです。だから、`...` が引数に当たります。

### 9.16.3 マクロの名前の付け方

このときの名前は、`[\namae]` のような英字も、`[\名前]` のような漢字も使うことができます<sup>25</sup>。しかし大文字と小文字は区別されます。

また記号や数字を含む名前も作ることができません。

<sup>23</sup> 何でも構いません。好きなものをつくって下さい。

<sup>24</sup> これってただの `center` 環境ですね

<sup>25</sup> あまり漢字はお薦めできません

## 9.17 ファイルを分けて L<sup>A</sup>T<sub>E</sub>X を使う方法

このガイドのように分量のあるものを作るとき、1つのファイルとして作業しているとエラーも見つけにくく、処理に時間がかかってしまいます。そこで「小さなまとまりごとに文書ファイルを作る」という考え方が出てきます。L<sup>A</sup>T<sub>E</sub>X では `\input{...}` を使って実現することができます。例として

```
\input{first.tex}
```

とすると、その場所に `first.tex` のファイルの中身を書いたことになります。

これを使って先程の `\KSU` を自分専用のファイルに入れましょう。まずエディタを立ち上げ、`macro.tex` というファイル名のファイルをつかって、その中に `\KSU` の定義を書き込んでおきます。そしてこの `macro.tex` というファイルを作業しているディレクトリに入れておきます。そして使いたい文書ファイルの中で次のように `\input{...}` を使ってこのファイルを読み込みます。

```
\input{macro}
```

そうすればその場所に `macro.tex`<sup>26</sup> のファイルの内容を書いたことになり、これ以降の場所で `\KSU` のマクロが使えるようになります。

## 9.18 標準以外のスタイルファイル

普通スタイルファイルは、`(j)article`, `(j)report`, `(j)book` の3つでしたが、以下にあげるような方法で、いろいろなスタイルファイルが使えます。

### 9.18.1 日本語 L<sup>A</sup>T<sub>E</sub>X 定番スタイル集の使い方

京都産業大学がライセンス契約しているインプレスの日本語 L<sup>A</sup>T<sub>E</sub>X 定番スタイル集の使い方を紹介します。このスタイル集を使うことによって、`jarticle` や `jreport` というスタイルだけでなく英文レター、はがき、時刻表、レポート用紙、アドレス帳、論文などのさまざまなスタイルの文書を作ることができます。使い方の例として英文レターの場合は次のようにします。

```
┌─── 入力 ───┐
\documentstyle{iletter}
\begin{document}
\begin{letter}
:
:
\end{letter}
end{document}
└──────────┘
```

のようにドキュメントスタイルの指定を変えてやります。またそれだけではなく、そのスタイルで使いやすいような命令、環境(ここでは `\begin{letter}... \end{letter}` のところです)が入っていますので、各情報処理教室の棚やロッカーにあるマニュアルを見て有効に利用して下さい。

<sup>26</sup> 例文では `.tex` を省略しました。同じファイルがなければ構いません。

## 第10章 AUCT<sub>E</sub>X

AUCT<sub>E</sub>X は Mule 上で動作し、T<sub>E</sub>X を書く作業を支援してくれます。この機能を利用することによって、L<sup>A</sup>T<sub>E</sub>X の文書を編集する作業がずいぶん便利になります。特にエラー修正の効率はかなり上がると思われます。これらの機能は初心者ほど効果があると思いますので、まずいきなり AUCT<sub>E</sub>X を使って L<sup>A</sup>T<sub>E</sub>X を試してみると言うのも良いかも知れません。但し Mule に機能を追加して利用していますので、Mule の操作に慣れている必要があります。

### 10.1 AUCT<sub>E</sub>X で L<sup>A</sup>T<sub>E</sub>X 生活が変わる

AUCT<sub>E</sub>X を使うとコマンドひとつで

```
\documentstyle[a4j]{jarticle}
\begin{document}

\end{document}
```

等を書いてくれます。カーソルは `\begin` の行と `\end` の行の間に置かれます。もちろん `document` 環境以外の任意の環境もコマンドひとつで書いてくれます。これで `\begin{}``\end{}` や `{}` の非対応のエラーから解放されます。

AUCT<sub>E</sub>X を使うと Mule の中からコンパイルとプレビューと印刷のコマンドを実行する事が出来るようになります。これで Mule と `kterm` を行ったり来たりする必要が無くなります。

AUCT<sub>E</sub>X を使うとエラーメッセージが日本語で表示されます。その上コマンドひとつで次々とテキストのエラー箇所にカーソルが移動します。これで意味が全く理解できなかった T<sub>E</sub>X のエラーメッセージからのデバッグ作業の苦痛は取り除かれます。

以上は AUCT<sub>E</sub>X の機能のほんの一部ですが、それらを使うだけでも L<sup>A</sup>T<sub>E</sub>X の編集作業が地獄から天国に一変する事でしょう。それではまず 10.2 で AUCT<sub>E</sub>X を起動してから、10.3 以降でこの節で紹介したすばらしいコマンド達の使い方を説明します。これらのコマンドはいわゆる必修項目です。といってもそのために覚える必要のあるコマンドはたったの 6 つ (実質 4 つ) です。その後、10.7 ではその他の便利なコマンドをざーっと紹介します。

### 10.2 AUCT<sub>E</sub>X の起動

さて、AUCT<sub>E</sub>X モードの起動からはじめましょう。cc 環境では AUCT<sub>E</sub>X は cc2000 の Mule から使うことができます。拡張子が `.tex` のファイルを cc2000 の Mule で読み込むと、自動的に AUCT<sub>E</sub>X が起動するように設定されています。そして AUCT<sub>E</sub>X は原稿を見て適切なモード<sup>1</sup> に入ります。新しいファイルを呼び出した時など、その情報が無い場合は、デフォルトとして LaTeX モードが起動します。

---

<sup>1</sup> AUCT<sub>E</sub>X には LaTeX モード、JTeX モード、JSliTeX モード、TeX モード等があります。

## 10.3 C-c C-e

AUCT<sub>E</sub>X のコマンドのほとんどは C-c で始まります．そしてそれに続く C-e の ‘e’ は ‘environment の e’ と覚えましょう．このコマンドは L<sup>A</sup>T<sub>E</sub>X の様々な環境、つまり `\begin{なんとか}\end{なんとか}` を対話的にセットしてくれます．

具体的な作業に進みましょう．新しい T<sub>E</sub>X の原稿の書き始めには、まず document 環境をセットするためにこのコマンドを入力することになるでしょう．すると AUCT<sub>E</sub>X が

```
Master file: (default this file)
```

と聞いてきます．これは新規ファイルに対して最初にコマンドを入力した時だけ聞いてくる質問で、10.7 で説明する分割編集の際に参照される情報です．ここではとりあえず <return> キーだけを押しおきましょう．すると、次に

```
Environment type: (default document)
```

と聞かれるので、使いたい環境名をタイプします．document 環境をセットしたいので、

```
Environment type: (default document) document
```

とタイプします．

丸カッコの中に「デフォルト ( default )」として document が用意されている旨が表示されています．この様に AUCT<sub>E</sub>X ではコマンド入力時には、大抵、良く考えられた「デフォルト」が用意されています．そのデフォルトを採用する場合には、何もタイプせずに <return> キーを押すだけで良いのです．以降では、積極的にこの機能を使ってゆく事にします．

先に進みましょう．document 環境をセットするための質問はまだ続いています．

```
Document style: (default jarticle)
```

ここで style 名をタイプします．デフォルトは jarticle になっています．デフォルトを採用する場合は単に <return> キーを押しましょう．jreport が良ければそうタイプして <return> キーを押します．最後に Options を質問してきますので、A4 の日本語なら a4j とタイプします．さらに原稿中で PostScript の図を取り込むなら、に続けて epsbox とタイプしておきます．

```
Options: a4j, epsbox
```

これで document 環境がセットされます．

他の環境もセットしてみましょう．例えば equation 環境なら

Environment type: (default itemize) equation  
label: eq:abel

のように対話が進みます。他にも, table 環境なら

Environment type: (default equation) table  
Float to: htbp  
Caption: すごい表  
Label: tab:wao!  
Center: (y or n) y  
Psition:  
Format: |c||llll|

のように AUCT<sub>E</sub>X とユーザーの間で対話が進みます。

## 10.4 C-c {

C-c { で}がセットされます。これで L<sup>A</sup>T<sub>E</sub>X で多用される中括弧の閉じ忘れのデバッグ作業から解放されます。

## 10.5 C-c C-c

このコマンドを使うことで, Mule の中からコンパイル, プレビュー, 印刷, スペルチェック等のコマンドを実行する事が出来ます。‘c’ は ‘command の c’ です。

L<sup>A</sup>T<sub>E</sub>X の原稿の編集が一段落したら, 原稿を L<sup>A</sup>T<sub>E</sub>X に通します。さあ, コマンド C-c C-c を入力しましょう。原稿を保存せずにこのコマンドを入力した時には, まずセーブするかどうかを聞いてきます。

Save file? /home/kyoin1/matsuura/tex/foo.tex ? (y or n) y

y とタイプすると

Command: (default jLaTeX)

の様に AUCT<sub>E</sub>X は聞いてきます。上の例では(default jLaTeX) となっています。つまり, jlatex コマンドがデフォルトに設定されています。デフォルトで良ければ何もタイプせずに<return>を押します。コンパイルを実行すると

Type ‘C-c C-l’ to display results of compilation.

と教えてくれます。コンパイルの様子を見たければ AUCT<sub>E</sub>X の言う通り C-c C-l と入力します。現在走らせているコンパイルなどを途中で中止させたい場合には C-c C-k と入力します。

コマンドは jLaTeX 以外にも多数用意されていますが、一つ一つのコマンドのスペルを覚える必要はありません。Command: と聞かれている時にスペースキーを押すと、全コマンド一覧の表示とコマンドの補完を同時に行なってくれます。その中から目的のコマンド名を探して、Command: の行にタイプして<return>を押します。AUCT<sub>E</sub>X はデフォルトの値を適切に選んで設定してくれるので、コマンドをタイプする事なく<return>キーを押す事が一番多いでしょう。View やPrint を選んだ時には、AUCT<sub>E</sub>X は実際に実行するコマンドを確認してきます。View の場合のやりとりを見てみましょう。

```
Command: (default View)
```

L<sup>A</sup>T<sub>E</sub>X のエラーやワーニングが無ければデフォルトが View になっている事でしょう。その場合、<return>キーを押します。

```
View command: xdvi foo
```

の状態です。そのまま<return>キーを押せばxdvi が起動します。例えば B4 用紙サイズで表示したい場合には、ここで

```
View command: xdvi -paper b4 foo
```

のようにコマンドに修正を加えてから<return>キーを押します。また、Print の場合は

```
Command: (default View) Print
```

```
Printer: (default cspr01)
```

のようにプリンターの名前を聞いてきます。印刷させたいプリンターが cspr01 でない場合はプリンター名をタイプします。次に実行されるコマンドを確認してきます。

```
Print command: jdvi2kps -d 400 foo | lpr -Pcspr01
```

問題なければそのまま<return>を押します。

## 10.6 C-c ‘

さあ天国に行きましょう。デバッグコマンドです。このコマンドは覚える必要さえありません。なぜならコンパイル中にエラーが見つかった場合、AUCT<sub>E</sub>X は親切なことに“エラーがあったよ。C-c ‘を押してね。”と教えてくれます。次のエラーに進むときも C-c ‘です。’ではなく‘であることに注意しましょう。

## 10.7 その他の機能

前節までの機能で  $\LaTeX$  生活環境は格段に向上します。ですから初めて  $\text{AUCTeX}$  を使おうという人はこの節をざっと目を通すだけにするか、あるいは読むのをやめてしまっても良いでしょう。

### 10.7.1 部分的なコンパイル

$\text{AUCTeX}$  を使うと部分的にコンパイルをすることができます。複雑な `equation` 環境や `table` 環境になると、コンパイルとプレビューを何度も繰り返しながら修正する事になります。その度に毎回ドキュメント全体をコンパイルするのはばかばかしいですね。そこで適当な領域だけを部分的にコンパイルすることになります。コンパイルしたい領域の先頭で `C-<Space>` を入力して、その領域の後尾の次の所にカーソルを置いて `C-c C-r` と入力します。するとコマンド入力待ちになるので、前述の手順 (10.5 節参照) に従ってコンパイルします。複雑な `equation` 環境や `table` 環境等を部分コンパイルしたいときには、部分コンパイルしたい環境の中にカーソルを置いた状態で `C-c .` と入力するだけで領域としてその環境が設定されます。そこで続けて `C-c C-r` と入力します。他にも、編集している節を領域として設定するためのコマンド `C-c *` があります。これも続けて `C-c C-r` と入力する事で、部分コンパイルをすることが出来ます。さらに編集しているバッファ全体を部分コンパイルするためのコマンド `C-c C-b` もあります。部分コンパイルの結果をプレビューするときも `C-c C-b` または `C-c C-r` と入力します。おそらく `View` がデフォルトに設定されているので、`<return>` を押せば `xdvi` が起動します。部分コンパイルの結果を印刷したければ `Print` とタイプして `<return>` を押します。

### 10.7.2 ドキュメントの分割編集

$\LaTeX$  のドキュメントを複数のファイルに分けて編集していく方法がよく取られます。 $\text{AUCTeX}$  を使うとその手の  $\LaTeX$  のドキュメントの管理が楽になります。ここでは例として、原稿全体は親ファイルと子ファイル (下の例では `section1.tex`, `section2.tex`, `section3.tex`) から構成されるとします。まず親ファイル (`parent.tex`) を作ります。10.3 節で説明したように、`document` 環境を入力する時に `master file` の名前を聞いてきますが、ここではそのまま `<return>` を押します。そしてそのファイルには

```
\documentstyle[a4j]{jarticle}
\begin{document}
\input{section1}
\input{section2}
\input{section3}
\end{document}

% Local Variables:
% mode: japanese-latex
% TeX-master: t
% End:
```

のように子ファイルを読み込むコマンドを書いておきます。次に子ファイルを作ります。新規に小ファイルを開いて最初に  $\text{AUCTeX}$  のコマンドを入力する時にも、例のごとく `master file` の名前を聞いてきます。ここでは、次のように親ファイル名の (拡張子 `.tex` は付けずに) フルパスをタイプします。

Master file: (default this file) ~/tex/parent

こうしておけば、いちいち親ファイルに戻らずに、子ファイルを編集している状態で C-c C-c と入力することで、ドキュメント全体のコンパイルが出来るようになります。そして編集中の思考錯誤時には、分割して編集しているメリットを生かして C-c C-b で子ファイルだけの部分的コンパイルを行なうとよいでしょう (10.7.1 参照)。子ファイルを編集中に親ファイルを見なくなった時には C-c ^ と入力すれば親ファイルが開かれます。

### 10.7.3 アウトラインマイナーモード

10.7.2 の様にドキュメントを分割して作成する方法の他にも、ひとつのファイルに全ての文章を書く方法もあります。しかしそうすると「さて第 2 章第 3 節を修正しようか」という時に、そこを画面に表示させるために C-v を何度も押して...おととと、行き過ぎた！<ESC> v, なんてことになりそうです。AUCT<sub>E</sub>X でアウトラインマイナーモードを使えばそんな面倒を避けられます。

それでは<ESC> x outline-minor-mode と入力してアウトラインマイナーモードに入りましょう。アウトラインマイナーモードに入ったら、最初に C-c C-o C-t と入力して文章部分 (ボディ) を隠しておきましょう。\\chapter や\\section の行 (ヘッダ行) だけが表示されます。この状態で、編集したい節の\\section の所までカーソルを持っていておいて C-c C-o C-e と入力すると、その節の隠されていた文章部分だけが表示されます。その節の編集を終えたら C-c C-o C-c と入力して再び文章を隠しておきます。ちなみに原稿全てのボディを表示させるコマンドは C-c C-o C-a です。

編集集中での各節を部分コンパイルする時は C-c \* を使って領域を指定しましょう (10.7.1 参照)。コマンドは上に挙げた他にも色々あります。その一覧を表 10.1 に示しておきます。

### 10.7.4 L<sup>A</sup>T<sub>E</sub>X マクロの入力

AUCT<sub>E</sub>X を使うと様々な L<sup>A</sup>T<sub>E</sub>X マクロを完全に覚えていなくても使う事ができるようになります。これで、L<sup>A</sup>T<sub>E</sub>X の教科書が横に無くてもある程度 L<sup>A</sup>T<sub>E</sub>X のコマンドが使えるようになります。その機能の一つが L<sup>A</sup>T<sub>E</sub>X マクロの補完です。補完といえば、<Tab>キーを使った tcsh の補完機能がお馴染みでしょう。AUCT<sub>E</sub>X での L<sup>A</sup>T<sub>E</sub>X マクロの補完のコマンドは<ESC> <Tab> です。候補が複数あると補完は途中で止まります。その場合は、その続きを何文字か入力してさらに<ESC> <Tab> を入力します。候補がひとつになるまでこれを続けて、ひとつになった所で L<sup>A</sup>T<sub>E</sub>X マクロが最後まで補完されます。マクロを対話式に入力する方法を使うことで、覚えていない L<sup>A</sup>T<sub>E</sub>X マクロを入力することもできます。C-c C-m で対話的に L<sup>A</sup>T<sub>E</sub>X マクロの入力を求められます。マクロ名を途中までしか覚えてなくても、<Tab>キーや<Space>キーを押せばマクロ名を補完してくれたり、マクロ候補一覧を表示してくれたりするので安心です。\\chapter{}や\\section{}も L<sup>A</sup>T<sub>E</sub>X マクロなのですが、AUCT<sub>E</sub>X ではこれらの入力には別のコマンドが割り当てられています。C-c C-s がそれで、以下のように対話しながら、\\chapter{}や\\section{}の入力を行いません。

```
Select level: (default section) chapter
What title: すごい章
What label: cha:wao!
```

表 10.1 アウトラインマイナーモードのコマンド一覧

key	binding
C-c C-o C-a	すべてを表示する
C-c C-o C-t	すべてのボディを隠す
C-c C-o C-o	カーソルの置かれている節以外のボディを隠す
C-c C-o C-q	カーソルの置かれている節のレベル以下をすべて隠す
C-c C-o C-e	カーソルの置かれている節を表示する
C-c C-o C-c	カーソルの置かれている節を隠す
C-c C-o C-l	カーソルの置かれている節以下のボディを隠す
C-c C-o C-k	カーソルの置かれている節以下のヘッダ行を表示する
C-c C-o C-d	カーソルの置かれている節以下をすべて隠す
C-c C-o C-s	カーソルの置かれている節以下をすべて表示する
C-c C-o TAB	カーソルの置かれている節直下のヘッダ行を表示する
C-c C-o C-b	同じレベルで一つ前の節のヘッダ行に移動する
C-c C-o C-f	同じレベルで一つ後の節のヘッダ行に移動する
C-c C-o C-u	一つ上のレベルの節のヘッダ行に移動する
C-c C-o C-p	カーソルの置かれている所より前方で、現在表示されている最も近いヘッダ行に移動する
C-c C-o C-n	カーソルの置かれている所より後方で、現在表示されている最も近いヘッダ行に移動する
C-c C-o C-h	コマンドヘルプを表示する

### 10.7.5 複数行のコメントの付け外し

AUCTeX を使うと連続した多数行にわたってコメント記号% を一度に付ける事ができます。コメント記号をつけたい領域の先頭で C-<Space> , 後尾の次の行で C-c ; とすればその領域にわたって行頭にコメント記号が付けられます。コメント記号を外したい場合には、その領域の先頭で C-<Space> , 後尾の次の行で C-u C-c ; とすればコメント記号が削除されます。

### 10.7.6 書体の指定

AUCTeX を使って書体設定の一括入力ができます。例えば bold 書体のコマンドを入力すると{\bf }とタイプされます。イタリック書体\it, emphasized 書体\em そして slanted 書体\s1 についてはイタリック補正もセットされるという芸の細かさです。

一旦書体指定をセットした後で書体指定を別の書体に変更したくなった場合には、C-u C-c C-f <KEY> と入力します。<KEY> には変更後の設定したい書体に対応する、C-b C-i C-r C-e C-t C-s C-c のいずれかが入ります。

### 10.7.7 数式モードの支援

AUCTeX を使うと数式モードで利用される記号の入力が簡単になります。C-c ~ で LaTeX-math-mode になります。再び C-c ~ で LaTeX-math-mode から抜けます。数式モードでは「'」(逆シングルクォート)と一文字の入力で、数式環境でよく使われる記号が展開されます。例えば ' a と入力すると\alpha がセッ

表 10.2 書体指定をセットするコマンド一覧

<b>bold</b>	C-c C-f C-b
<i>italics</i>	C-c C-f C-i
roman	C-c C-f C-r
<i>emphasized</i>	C-c C-f C-e
typewriter	C-c C-f C-t
<i>slanted</i>	C-c C-f C-s
SMALL CAPS	C-c C-f C-c
書体設定の消去	C-c C-f C-d

表 10.3 数式モードでの記号入力

### ギリシャ文字

$\alpha$	(alpha)	a
$\beta$	(beta)	b
$\delta$	(delta)	d
$\epsilon$	(epsilon)	e
$\phi$	(phi)	f
$\gamma$	(gamma)	g
$\eta$	(eta)	h
$\kappa$	(kappa)	k
$\lambda$	(lambda)	l
$\mu$	(mu)	m
$\nabla$	(nabla)	N
$\nu$	(nu)	n
$\omega$	(omega)	o
$\pi$	(pi)	p
$\theta$	(theta)	q
$\rho$	(rho)	r
$\sigma$	(sigma)	s

### 記号

$\tau$	(tau)	t
$\upsilon$	(upsilon)	u
$\chi$	(chi)	x
$\psi$	(psi)	y
$\zeta$	(zeta)	z
$\Delta$	(Delta)	D
$\Gamma$	(Gamma)	G
$\Theta$	(Theta)	Q
$\Lambda$	(Lambda)	L
$\Psi$	(Psi)	Y
$\Pi$	(Pi)	P
$\Sigma$	(Sigma)	S
$\Upsilon$	(Upsilon)	U
$\Phi$	(Phi)	V
$\Omega$	(Omega)	O

$\rightarrow$	(rightarrow)	C-f
$\leftarrow$	(leftarrow)	C-b
$\uparrow$	(uparrow)	C-p
$\downarrow$	(downarrow)	C-n
$\leq$	(leq)	<
$\geq$	(geq)	>
$\sim$	(tilde)	~
$\infty$	(infty)	I
$\forall$	(forall)	A
$\exists$	(exists)	E
$/$	(not)	!
$\in$	(in)	i
$\times$	(times)	*
$\cdot$	(cdot)	.
$\subset$	(subset)	{
$\supset$	(supset)	}
$\subseteq$	(subseteq)	[

$\supseteq$	(supseteq)	]
$\backslash$	(backslash)	\
$\setminus$	(setminus)	/
$\cup$	(cup)	+
$\cap$	(cap)	-
$\langle$	(langle)	(
$\rangle$	(rangle)	)
exp	(exp)	C-e
sin	(sin)	C-s
cos	(cos)	C-c
sup	(sup)	C-^
inf	(inf)	C-_
det	(det)	C-d
lim	(lim)	C-l
tan	(tan)	C-t
$\hat{\phantom{x}}$	(hat)	^
$\vee$	(vee)	v

トされます。表 10.3 が数式モードのコマンド一覧です。

## 10.8 最後に

AUC<sub>T</sub>E<sub>X</sub> にはまだ他にも幾つかのコマンドがあります。また、カスタマイズについては説明しませんでした。それらについては AUC<sub>T</sub>E<sub>X</sub> のパッケージに付属の英語のマニュアル<sup>2</sup> や info ファイルを参照して下さい。また、X 環境で Mule を使っている場合に限られますが、AUC<sub>T</sub>E<sub>X</sub> は Mule の (というより Emacs 19 の) 拡張機能の一つであるメニューパー機能に対応しています。(Japanese-) L<sup>A</sup>T<sub>E</sub>X-mode に入ると LaTeX という項目が追加され、そこに AUC<sub>T</sub>E<sub>X</sub> のコマンドのメニューが登録されています。

---

<sup>2</sup> dviman というコマンドを用意しています。とりあえず、ターミナルで `dviman -l` とタイプしてみてください。

# 第11章 どこからでも telnet

ネットワークのある暮らしはなかなか便利なものです。自分宛のメールをチェックしたり、メッセージを誰かに送ったりしたいと急に思い付いても、学内にいる限りどこか近くの空いている情報処理教室などに飛び込めば何とかできます。ところで、あなたが大学のキャンパスを離れて、例えば下宿や自宅に帰ってしまったら、今度大学に行くまでメールのやりとりはお預けにしないと駄目なのではないでしょうか？

そんな事はありません。ネットは繋がっているのです。極端なことを言うと、コンピュータがあって、それをネットワークを経由して cc 環境につながるようにさえすれば、あなたはそこから cc 環境を利用することができるのです。

cc 環境が提供するネットワークサービスは telnet による UNIX コンピュータへの login を基本としています。つまり、どこからでも cc 環境マシンに telnet して login できさえすれば、あなたは cc 環境のネットワークサービスを大学にいるときと大して変わり無く利用することができるのです。

ここでは学外、特に下宿や自宅から cc 環境のネットワークサービスを利用する方法を説明します。

## 11.1 スロースタート

特に慣れない人にとって、ここからの道のりは結構遠いかもしれません。ゆっくり説明していくことにしましょう。

### 11.1.1 どこから telnet？

あなたが学外から cc 環境を利用できる場合は、おそらく以下のいずれかとなるでしょう。

- 他の大学などのインターネットに接続されたコンピュータの前に座っている。
- インターネット・カフェなどと呼ばれる場所のパソコンの前に座っている。
- 下宿や自宅のパソコンの前に座っている。

はじめから二つは共にインターネットに常時接続されたコンピュータだと思われます。また、見方を変えると誰かが管理者となってインターネットへの接続の面倒を見ているコンピュータだとも言えます。逆に最後の場合は、あなたが管理者となって自分で自分のコンピュータをネットワークに接続する必要があります。つまり上の分類は以下のように言い替えることができます。

- 誰かが管理しているコンピュータを使って telnet する。
- 自分のコンピュータを自分で telnet できるようにして、そこから telnet する。

この章ではそれぞれの場合について説明しています<sup>1</sup>。

---

<sup>1</sup> 他の可能性、例えば学内の実験室や研究室のコンピュータから利用する場合もあるでしょう。しかしそれについては状況が個々に違い過ぎる為、ここでは説明しません。適当に以下の記述から類似点を手掛かりに情報を拾い上げてください。また本学 Web ページにも紹介されていますのでそちらも参照してください。

### 11.1.2 どこへ telnet?

インターネットにつながっているコンピュータがある所なら、世界中のどこからでもあなたは cc 環境のマシンに telnet できるはずですが、しかしそのためには cc 環境のホストコンピュータの名前、それもインターネット上で通用するワールドワイドな名前を知っていないといけません。

京都産業大学のネットワークを表すインターネット上での名前は kyoto-su.ac.jp で、これをドメイン名と呼んでいる事は既に 114 ページの 第 5.2 節のはじめの方で説明しました。

今まで学内のコンピュータはホスト名だけで指定していました。ホスト名の後ろにピリオドで区切って、このドメイン名を付け足した名前こそが、インターネット上のどこからでも通用する名前なのです。具体的には学内で cc2000 と呼んでいたコンピュータは cc2000.kyoto-su.ac.jp と表記します。

cc 環境の中で telnet を受けてサービスを提供しているコンピュータは以下のものです。

---

機種名	ドメイン付きホスト名
Enterprise3000	cc2000.kyoto-su.ac.jp

---

計算機センター 1 階に設置の Sun 社製コンピュータ。

---

もしもどこかで telnet する場合、ホスト名としては上記のコンピュータをドメイン付ホスト名で指定してください。ドメイン名を省略できる場合もありますが、その判断が付かない場合はドメイン付ホスト名を使うように心掛けていると良いでしょう。

### 11.1.3 これ以降の読み方

先が少々長くなりますので読み進む順番をあらかじめ示しておきます。

- 誰かが管理してくれているコンピュータを使って telnet する場合は、次の第 11.2 節に進んでください。
- 自分のコンピュータを自分で telnet できるようにして、そこから telnet する場合は、302 ページの第 11.3 節に進んでください。
- 特に自分がモデムを使った PPP 接続などに慣れていると思える人は 339 ページの第 11.8 節に進み、拾い読みすれば足りるかもしれません。  
ただし、ダイヤルアップ IP 接続を利用する場合は、307 ページの第 11.4.1 節に「利用上の注意と制限」がありますので、この部分だけは必ずしっかりと読んでください。

## 11.2 誰かが管理してくれているコンピュータから

結論は簡単です。その場所では、どのようにして telnet したら良いのかを調べてください。調べる方法が判らなければ、その場所のコンピュータの管理者などに連絡をとって「自分の大学のコンピュータがインターネットに接続されているので、そこに telnet をする」為に必要な操作方法を教えて貰う事になるでしょう。

非常に馬鹿な事に思えるかもしれませんが、これは大切なことです。つまりあなたが、もしその場所で telnet できそうに思える方法を知っていたとしても、その方法が管理者にとって望ましい方法がそうで無いか問題なのです。その場所ではその場所の管理者の決めたルールに従うことが最も重要です。知っているからやってみる、できるからする、という態度に単純に走るのは悪い考え方です。常に人と共有する機械を使っているという事を忘れないようにしましょう。

### 11.2.1 誰かが管理してくれているコンピュータからの接続に対するトラブル対策

以下にこのような場合で起こりそうなトラブルについて、その原因と解決策を示しておきます。少し技術的に細かな事が書かれている場合もあるでしょう。あなたがそれを読んで判らない場合でも、その管理者の参考になるかも知れないと思うからです。

なお、対策として設定を変えたら動作するという記述があると思いますが、その場合は常にその場所の管理者に設定を変更しても良いかどうか確認してから実行する必要があります。

#### telnet したがつながらない

cc2000 ではなく、cc2000.kyoto-su.ac.jp と最後まで入力していますか？学内からは cc2000 でつながっても学外からは全部入力が必要です。cc2000.kyoto-su.ac.jp を相手に telnet してもつながらない場合は、使おうとしているコンピュータの管理者に聞いてみるのがいいでしょう。もしかするとそのような使い方を認めない運用をされているのかもしれない。

#### アルファベットは表示されるが漢字部分の表示がおかしい

お使いの telnet ソフトの文字コード<sup>2</sup> が正しく設定されていないようです。コンピュータの管理者に了解を得たうえで EUC コードにセットしてください。

#### Mule を使おうと思うのだがうまく動作しない

87 ページの第 4.5.2 節、特に「非 X ウィンドウ環境」での起動方法とエラー対策を参考にしてください。

#### Mule を使おうと思うのだが C-s , C-c , <Delete>などが動作しない

お使いの telnet ソフトの設定に問題があるようです。コンピュータの管理者に相談してください。

---

<sup>2</sup> 148 ページ、第 5.5.3 節参照

## 11.3 あなたのパソコンから

ちょっとあなたの部屋の周りを見回してください。もしあなたが使えるパソコンと、電話回線がそこにあるのなら、少しお金を出して準備すれば、あなたのパソコンから電話線経由で cc 環境を利用することができるようになります。

あなたのパソコンから cc 環境を利用するには、とにかくあなたのパソコンと cc 環境のコンピュータを電線で結ばなければなりません。何故ならコンピュータの中では信号が電気に変えられて流れているからです。電話線だって電線なのですが、この電線に例えばコンピュータの電気信号を流すわけには行きません。しかしコンピュータの信号を、音の信号に変えてから電話線に通す事はできます。電話線を通したからそれをまたコンピュータの信号に戻せば良い訳です。音に変えられた信号が電話線を通り抜けて、あなたのパソコンと電話の向こうのコンピュータの間を往き来しているところを想像してください。

これ以降は下宿や自宅のあなたのパソコンを使って、大学のネットワークサービスを利用する方法を説明します。勿論旅行先にノートパソコンを持っていても同様のことができます。

でもちょっと待って!

なんだか良さそうな事ばかり書きました。でも公平に良くない事も書かなくてはなりません。今までこの冊子に書かれてきた事は、全て誰かが管理者となって使いやすい状態に管理されたコンピュータを使う場合に、最低限知らなければならないことについて書かれていました。しかし今度はあなたが自分のコンピュータの管理者となるのです。コンピュータに関する知識や、トラブルを解決する為の努力が、ある程度要求される事に注意してください。それがどうしても嫌なら、誰かに管理されたコンピュータを使わせて貰うことをお勧めします。つまり大学に行けば良いという事です。あなたはそれができる立場なのであります。

### 11.3.1 準備段階

あなたが自分のコンピュータから電話線経由で telnet するためには、以下のものが必要になります。

#### コンピュータ

まずはコンピュータがないと話になりません。どのような種類のコンピュータでも、一般的にパソコンショップで売っているパソコンならば大抵問題なく電話線経由で telnet することができます。購入前には誰かよく知っている知人などにアドバイスを貰うのが良いかもしれません。学内ローカルニュースグループの一つ、sandai.questionなどを覗く、または質問してみるのも最新の情報を集める一つの手段です。

#### 自分が使える電話回線

これも無いと話になりません。なお、携帯電話、PHSでの通信も可能ですが、ここでは説明しません<sup>3</sup>。NTTの(無線ではない)電話回線を使う場合についてだけ説明します。それから以下の点について確かめておいてください。

- あなたの電話回線はプッシュホン回線ですか？ダイヤル回線ですか？<sup>4</sup>

まずこれを確認しておいてください。このすぐ後で必要になります。毎月の料金明細を見たり、NTTに問い合わせるのが最も確実です。自分で判断するなら、指で回すダイヤルがついている電話機の場合

<sup>3</sup> 技術的に少々難しくなる上に、余り一般的ではないと思えるからです。雑誌などを見ると結構載っていますのでトライしてみるのも楽しいですよ。

<sup>4</sup> それにしてもこのプッシュホン回線、ダイヤル回線という呼び名はおかしなものです。トーンダイヤル、パルスダイヤルという呼び方もあります。要するに相手先番号を指定するのに、音色(トーン)を使うのか、断続的な信号(パルス)を使うのかを指していることには変わりはないのです。ボタンを押し(プッシュ)で音色を出し、円盤(ダイヤル)を回してパルスを作る。なるほど。でも押しボタンでパルスを作ることもできるし、そもそもトーンでダイヤル(目盛り盤)って何なんだ？

合<sup>5</sup> はまず間違いなくダイヤル回線です。押しボタン式の電話機でも、電話を掛ける時に数字のボタンを押したら、「ピッポッパッ」と言わずに「プツプツ」というノイズ音が聞こえるようならダイヤル回線の可能性が高いです。「ピッポッパッ」と聞こえてくればプッシュホン回線である可能性が高いですが、ダイヤル回線でも電話機が「ピッポッパッ」と代返している可能性もあるので要注意です。

- 回線の口がモジュラジャックになっていますか？

電話機から出ている電話線の先端を見てください。そこには 5mm × 8mm 程度の大きさの、透明なプラスチックになっていませんか？このタイプの差し込み口のことをモジュラジャックと呼んでいます。

もしそうなっていない場合は NTT に連絡してモジュラジャックの取り付け工事を頼む必要がありますが、それには工事費がかかりますので注意してください。

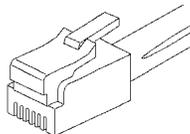


図 11.1 モジュラジャック

- その電話回線は ISDN と呼ばれるタイプではありませんね？

ISDN 利用者はこのガイドの利用者には多くないと判断して、ISDN 利用者のための説明はここでは行ないません。これから行なわれる説明は全て ISDN でない電話回線に絞って書かれており、かなりの相違があります。ISDN 利用者はその相違を自分で理解して読み替えてください<sup>6</sup>。

- その電話回線はホームテレホンと呼ばれるタイプではありませんね？

例えば「外線」などと書かれているボタンを押してからダイヤルしてはじめて外部に電話できる種類の電話などがそうです。この場合にはいろいろと複雑な問題が出がちです。大抵の場合、対処方法はあるのですが、電話機のメーカーなどによってまちまちなので、ここに一概にまとめて書くことができません。

その電話回線からの接続をあきらめるのが簡単ですが、よく知っていそうな知人に聞くか、学内ローカルニュースグループの一つ、sandai.question などに質問してみるなどして解決策を捜してみると良いでしょう。

- その電話回線にはキャッチホン契約をしていませんか？

キャッチホン契約をしていると、コンピュータが電話回線を使って大学と接続している時に誰かが電話を掛けてきた場合、あの「ぶちっ」という音によってコンピュータの電話接続が切れてしまいます。キャッチホン契約をやめるか、途中で切れてしまうのをあきらめるか、もしくは NTT が行なっているパソコン通信対応のキャッチホンサービス、「キャッチホン II<sup>7</sup>」などで対応してください。

## モデム

モデムとはコンピュータと電話回線を接続するための機器です。このモデムこそが、コンピュータの信号を音に変えて電話線に流してくれるのです。

<sup>5</sup> 俗に言う黒電話など、少し古い電話機がそうですね。

<sup>6</sup> 計算機センターの ISDN 接続設備自体は用意されています。あなたが ISDN のデジタル通信機能つき TA を用意しているのなら、それを使うことができます。

<sup>7</sup> NTT の WWW ページ (<http://www.ntt.co.jp/>) で紹介されています。

(1998 年 3 月時点では <http://web.hqs.cae.ntt.co.jp/shop/annai/annai.html>)

モデムが最初からコンピュータに内蔵されている機種もありますが、そうでない場合の方が多いと思います。付いていない場合は商品ですから買わなければなりません。モデムと一口に言っても、性能別、またコンピュータに合わせて様々な種類があります。

まずモデムの性能は一般的には通信速度で測られます。bps<sup>8</sup> という単位で表現されますがこれが 9.6Kbps (9600bps) もしくはそれ以上の値のものを推奨します。

もしも新規にモデムを購入するのであれば、33.6Kbps (33600bps) もしくはそれ以上<sup>9</sup> の値のものを強く推奨します。通信速度が遅いと、必要な作業をするための時間が長くなってしまい、使えば使うほど電話代がより多く掛かることとなります。一概には言えませんが、例えば 14.4Kbps のモデムでは 28.8Kbps の半分の速度ですので、同じ作業をするのにおよそ倍の時間がかかり、電話代も倍になるといふ具合です。

次にコンピュータの種別によって、主として以下の二種類の形のモデムが売られています。

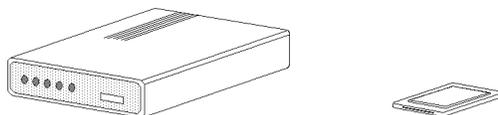


図 11.2 典型的なモデムの外形 (左:外置き型 右:カード型)

右のカード型がちょうどキャッシュカードの大きさと厚さ 3mm 程度、左の外置き型はそれとの比較で想像してください。つまりコンピュータの外に置いて電線で接続するものと、コンピュータの中に差し込んで接続するものです。後者は主にノートパソコンと呼ばれる種類のコンピュータで見られるものです。大抵のノートパソコンは前者の外置き型のモデムも使えますので、値段、利便性などを考えて選択すると良いでしょう。

接続の方法については基本的にはモデム購入時に製品に添付されるマニュアルに、どうやってコンピュータをモデムにつなぎ、そのモデムを電話回線につなぐのかが書かれているはずですが、詳細はそちらを参照してください。

いずれにしても購入前にそのモデムが自分のパソコンとつないで使えるものかどうかよく確認するのがお勧めです。

通信に必要なソフトウェア。

これについては機種、接続方法 (後述) などによって違いがありますので、後で別々に説明します。

### 11.3.2 ダイアルアップ IP 接続

これから本書で説明する接続方法はダイアルアップ IP 接続と呼ばれるものです。まず最初にあなたのコンピュータがダイアルアップ IP 接続を利用できるか、確認しておかなければなりません。通信速度が 9.6Kbps (9600bps) 以上の比較的高速なモデムを持っていて、以下のような条件に該当するコンピュータならばダイアルアップ IP 接続が利用できます。

- a1. 自分のパソコンは Windows95 で動いている。
- a2. 自分のパソコンは Macintosh である。

対して、通信速度が 2400bps 以下の比較的低速なモデムしか持っていないか、次のような条件の人は、筆者が検証できないため本書では説明できません。

<sup>8</sup> bit per second の略。一秒間に送信、または受信できるビット数。

<sup>9</sup> 56Kbps には二種類の規格がありますが、1998 年 3 月現在計算機センターが用意しているのは K56flex です。

- b1. 自分のパソコンは DOS で動いていて、Windows は使えない。
- b2. 自分が使おうと思っているのはパソコンではなく流行の携帯情報端末 (SHARP の Zaurus や NEC の Mobile Gear 等) だ。
- b3. 自分が使おうと思っているのは HP200LX もしくは通信機能つきワープロだ。
- b4. 自分のパソコンは Unix が動いている。

ここから先は、それぞれ以下の節に進んでください。環境ごとの設定の方法をそれぞれ説明します。

- a1. の人は 308 ページの第 11.5 節へ。
- a2. の人は 328 ページの第 11.7 節へ。
- b1., b2., b3., b4. の人は 339 ページの第 11.8 節などを参考に、持っている機器に添付されていたマニュアルをよく読んでトライしてください。

cc 環境と違って、各個人の環境というものには非常にばらつきがあって、一概には言えないことが多くあります。ここに書かれたことだけでうまく行かなかったとしても、個別の問題を少し解決してやるだけで何とかなる可能性もあります。うまく行かない場合などはインターネットやコンピュータ関連雑誌を読む、sandai.question などの学内ローカルニュースグループを覗く、またそこに質問するなどして見るのが良いでしょう。

### 11.3.3 ダイアルアップ IP 接続について

#### 用語の説明

今までの部分は何より専門的な知識の無いままで、概念的な理解をすることだけを目的に書かれていました。ところがダイアルアップ IP 接続をはじめとして、インターネットの流行にともなって多くの雑誌が自宅からダイアルアップ接続するための情報を載せはじめました。それらの雑誌で書かれている用語は、必ずしも適切なものではないかもしれませんが、だからと言ってこのガイドがそれとは違う独自の用語で書かれているのも困りものです。なにしろ熱心な人ほどその両方を読むのですから。

- ダイアルアップって何ですか？  
単純に訳して「電話を掛けて受話器を上げる」という事でしょうか。つまりコンピュータを使って電話を掛けさせる事を言います。電話を受けるのはやはりコンピュータ的なものです。多く登場した「ダイアルアップ IP 接続」というのは、つまり電話を掛けて、電話線経由で IP 接続するという事なのです。ところで短く「ダイアルアップ接続」と呼べば良さそうなものですが、IP 接続でないダイアルアップ接続も世の中にはある<sup>10</sup> ので注意が必要です。
- IP<sup>11</sup> って何ですか？ IP 接続って何ですか？  
IP というのは、ある種の通信規約の事ですが、難しいことはもうちょっと置いておいて、とにかくインターネットで使っているのと同じ通信のやり方なだと思ってください。つまり IP 接続というのは、インターネット的な方法でコンピュータもしくはネットワークをつないでしまうという事を意味しています。大学へのダイアルアップ IP 接続というのは、つまり電話線経由であなたのパソコンを大学のキャンパスネットワークの一員にするという事なのです。京都産業大学のキャンパスネット

<sup>10</sup> 京都産業大学の計算機センターは提供していません。

<sup>11</sup> あいびと読んでください。

ワークはインターネットに接続されていますから、つまりあなたのパソコンはインターネットの一員になったという事ですね。

- PPP<sup>12</sup> って何ですか？

PPP というのはネットワーク的な接続を実現するための、ある種の通信規約の事ですが、やはり難しいことは置いておきましょう。一般的にダイヤルアップで IP 接続をする時には、大抵の場合この PPP 技術を利用しています。そのためにダイヤルアップ IP 接続の事を「ダイヤルアップ PPP 接続」または短く「PPP 接続」と呼んだりする事があるようです。

- アクセスサーバって何ですか？

これも余り良い名前ではないように思います。ダイヤルアップ IP 接続のための設備を販売している企業が、その種の製品の事をこう呼んでいるのですが、余り一般的でもありませんし、意味的にも良くありません。(IP 接続という機能が追加された) ターミナルサーバと呼んでも良いと思うのですが、そうすると以前から売られていた telnet サービスを主体としたターミナルサーバと同一視されてしまうために、わざわざ新しい単語を用意したのでは無いかと思ってしまう。いずれ妥当な呼び名が定着するでしょうから、それまでこう呼んで過ごしましょう。

### より技術的な理解の為に

上の説明では省略して表現した部分を書いておきます。特に技術に興味のある人が、より正確な理解を得られるかもしれないと思うからです。

- PPP について。

PPP (Point to Point Protocol) 以外にも同様の機能を果たす実用化された別の規約、例えば SLIP (Serial Line IP) 等が存在します。しかし現在では PPP の方が、より普及しています。PPP 以外によるダイヤルアップ IP 接続も存在しますし、ダイヤルアップでない PPP 接続も存在します。PPP による IP でないネットワーク接続も存在します<sup>13</sup>。一般的には「PPP によるダイヤルアップ IP 接続」が最も普及しているために、PPP という用語はダイヤルアップ接続や IP 接続の周辺で曖昧に用いられる傾向があるように見えます。電話というものは端点と端点を結ぶ (Point to Point) ものだから、そこに PPP が適用できるのだ、という理解が最も妥当だと思います。そして PPP は IP を扱えるので、電話 + PPP で IP 接続ができるという訳です。

---

<sup>12</sup> ビービービーと読んでください。

<sup>13</sup> SLIP は名のとおり IP しか扱えません。

## 11.4 ダイアルアップ IP 接続の設定と利用法

これから各機種ごとの設定と利用法を個別に説明します。

### 11.4.1 利用上の注意と制限

ダイアルアップ IP 接続を利用する際に以下のことに注意してください。

長時間の回線占有に注意してください。

ダイアルアップ IP 接続を行なう人は非常に多く、今後も増え続けると思われます。そのためダイアルアップ IP 接続をしようとしても常に全ての回線が利用されており、話中ばかりで電話がなかなかつながらず、ほんの一通か二通メールを読み書きするだけなのに何時間もダイアルし続けなくてはならないなんて不合理です。

ダイアルアップ IP 接続のための回線は以下の三種類用意されています。あなたのパソコンの設定の変更は電話番号の部分だけで、他の設定は全てそのままどちらのタイプでも正しく動作するはずで

	電話番号 (内線)	最大接続速度	回線数	備考
タイプ I	075-705-3005 (なし)	64Kbps	46	デジタル回線でも接続可能
タイプ II	075-705-1502 (2990)	28.8Kbps	8	
タイプ III	075-705-1489 (2592)	14.4Kbps	5	

三種類あるのですから、タイプ II、III 側はせいぜい数分間の短時間利用にとどめるようにして、いつも空きを多く作るようにしましょう。長い時間の利用はタイプ I でお願いします。そうすることでみんな幸せな環境になるでしょう。ご協力をお願いします。

いずれにしても電話線は多くの利用者で使い回しているのですから、つながったのを幸い何時間も回線を占有して他の人がつながらなくなってしまうようでは困ります。どちらのタイプにしても混雑している時間帯、特にテレホーダイ<sup>14</sup> が適用できる時間帯などに、不必要に長時間接続して他の人に迷惑を掛けないように心がけてください。

このルールは紳士協定であり、特に制限も罰則も設けられてはいません。しかし最終的にルールを全体として守れず、電話回線が少数の利用者によって占有されてしまうようなことになった時には、様々な利用上の制限や罰則が設けられる可能性があります<sup>15</sup>。もちろん計算機センターは監視と罰則付きの運用を望んではいません。受信設備は非常に高価<sup>16</sup>で、少なくとも近い将来にどれだけ回線を用意すべきなのか、計算機センターも利用者と一緒に妥当な答を探しているのです。そのことを理解して、控え目な利用と紳士的な運用に御協力ください。

<sup>14</sup> NTT が提供しているサービスで、夜 11 時から朝 8 時頃までの近距離の特定電話番号に対する通話料金を月額固定にする。

<sup>15</sup> 例えば学内設備への接続以外できなくする、利用者あたりの月間接続時間に上限を設ける、一定時間に達したら強制的に回線を切断するなど制限が考えられます。管理と運用のために利用統計も取っていますから、占有が激しい場合には制限を加えるより先に、利用者ごとの月間利用時間合計を毎月掲示する方法も考えられます。

<sup>16</sup> 現在の製品では受信設備の追加には 1 回線当たり 10 万円以上掛かります。

## 11.5 Windows95 編

以下に Windows95 が動作するコンピュータでダイヤルアップ IP 接続を行う例を示します。

### 11.5.1 用意する物

#### ハードウェア

PC/AT 互換機や PC-98	Windows95 が動作していること。
モデム	内蔵されているコンピュータもあります。
モデムケーブル	(大抵の場合はモデムを買うと付属しています。)

#### 必要なソフトウェア

まずは接続のために。

Windows95 標準のダイヤルアップ機能。

接続してネットワークサービスを利用するために。

Telnet	Windows95 に標準添付しているが Tera Term 等を推奨。
Ftp	Windows95 標準添付。
Netscape あるいは Internet Explorer	WWW ブラウザ。最新の物を推奨。利用は無料。

#### あれば便利なソフトウェア

ネットワークサービスをより便利に利用するために。インストールの方法は説明しません。必要なら各ソフトに付属の説明書を読んでお使いください。

WS_FTP	FTP ソフト。
AL-Mail	電子メールソフト。最新の物を推奨。学生の利用は無料。
WinVN	ネットニュースソフト。最新の物を推奨。利用は無料。

### 11.5.2 インストール

マイコンピュータをダブルクリックします。開いたウィンドウから「ダイヤルアップネットワーク」を探してください。ダイヤルアップネットワークが無い人だけ次の作業を行ってください。ある人は読み飛ばして次へ進みます。



この枠内の説明はあなたのコンピュータで既に設定済みであるかもしれませんが。二重に登録するとおかしくなりますのでその場合読み飛ばしてください。特に最初から Windows95 がインストールされて販売されているコンピュータは設定済みである可能性が高いです。

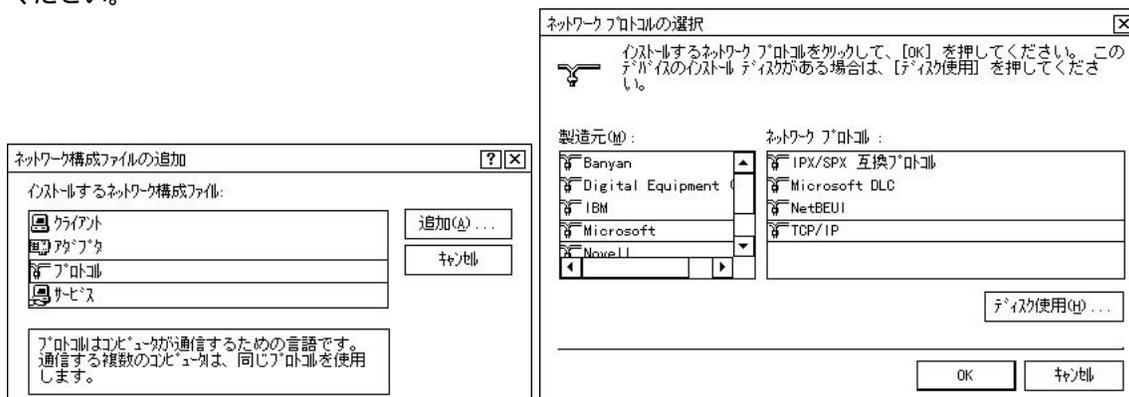
コントロールパネルからアプリケーションの追加と削除をダブルクリックします。出てくるウインドウの下の方にある「通信」をダブルクリックして、「ダイヤルアップネットワーク」の項目を確認します。左側の四角印にチェックがなければチェックします。



ここで、まだ入力していなければ コンピュータ名を入力を求めてきますので入力してください。コンピュータ名 / グループ名 / コンピュータの説明の 3 項目を聞いてきます。ダイヤルアップで京産大に繋げるのなら、この項目は京産大ネットワークに一切影響を与えません。自分の好きなように名前をつけてください。

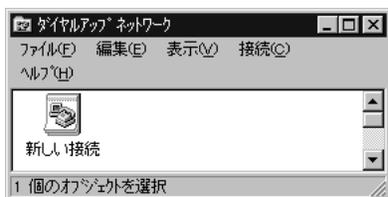
これが終わると再起動を聞いて来ますが、まだなくてもいいです。

次にコントロールパネルからネットワークをダブルクリックしてください<sup>17</sup>。そこに TCP/IP という項目があるか確認してください。TCP/IP がなければ追加ボタンを押して、プロトコル から Microsoft、TCP/IP を順にクリックして追加します。するとウインドウに TCP/IP が追加されます。TCP/IP が同じ名前でも 2 つ以上あると正しく動作しません。2 つ以上追加してしまったときは 1 つを残して削除してください。



ここまでできたらコンピュータを再起動してください。マイコンピュータに「ダイヤルアップネットワーク」が現れるはずですが。

マイコンピュータの「ダイヤルアップネットワーク」をダブルクリックして、「新しい接続」をダブルクリックします。



ここでモデムが設定されていない場合は、モデムのインストールが始まります。モデムのインストールは機種ごとに異なるので、モデムの説明書を読みながらインストールしてください。

モデムのインストール途中に次のような設定画面が出てきます。これはコントロールパネルのモデムを選択し、モデムのプロパティを選択して設定することもできます。お使いの電話にあわせて、市外局番、外線発信番号、ダイヤル方法にそれぞれ入力します。外線発信番号は0発信なら0を入れます。外線を掛ける時に何もせずに直接掛けられるのならここは空白にしておきます。ダイヤル方法はトーンかパルスか<sup>18</sup>を選択します。

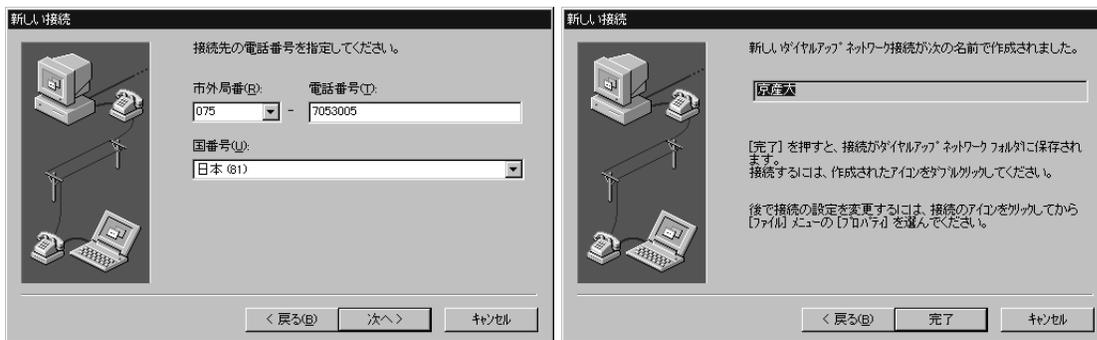


モデムのセットアップができれば、新しい接続として接続名を入力します。とりあえず京産大とでもしておきましょう。

<sup>18</sup> 302 ページ、11.3.1 をご覧ください。



次を選ぶと接続先の電話番号を聞いてきますので、  
 市外局番 075  
 電話番号 705-3005  
 国番号 日本  
 として「次へ」を選び、「完了」を選びます。



これでダイヤルアップネットワークの「新しい接続」横に「京産大」が作成されているでしょう。これを右クリックしてプロパティを選んでください。



現れた画面から「サーバの種類」のボタンを押します。図のようにチェックして「TCP/IP 設定」のボタンを押します。

京産大

情報

京産大

電話番号の入力

市外局番(R): 075 - 電話番号(P): 7053005

国番号(U): 日本 (81)

国番号と市外局番を使う(S)

接続の方法(N): MICROCORE MC288XL #2

設定(O)...    サーバーの種類(T)...

OK    キャンセル

サーバーの種類

サーバーの種類(S): PPP, Windows 95, Windows NT 3.5, インターネット

詳細オプション

ネットワークへのログオン(L)

ソフトウェア圧縮をする(C)

暗号化パスワードを使う(E)

使用できるネットワークプロトコル

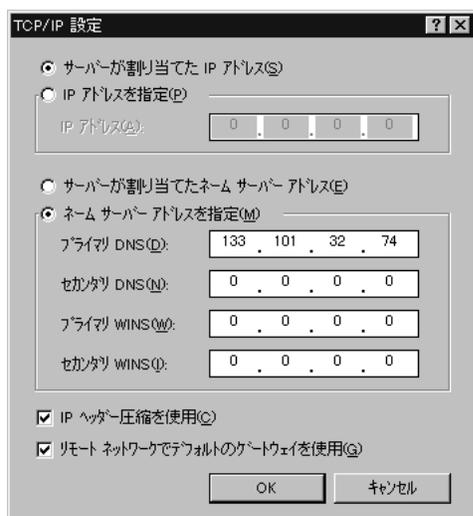
NetBEUI(N)

IPX/SPX 互換(I)

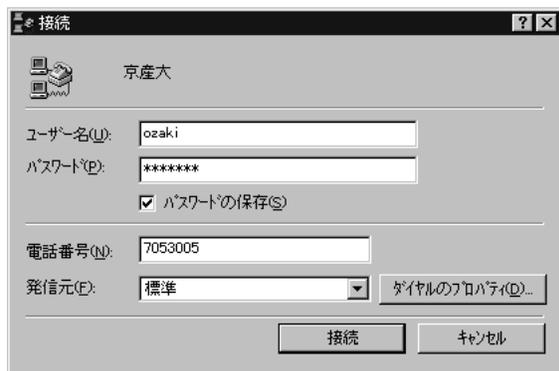
TCP/IP(T)    TCP/IP 設定(P)...

OK    キャンセル

TCP/IP の設定を図のように「サーバーが割り当てた IP アドレス」、「ネームサーバーアドレスを指定」、「プライマリ DNS 133.101.32.74」と設定して OK を押します。他のウィンドウも OK を押して閉じてください。これで大学に接続するための設定は終了です。

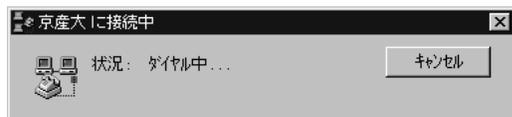


では京産大のアイコンをダブルクリックしてみましょう。次のような画面が出てきますのでいつも学校で使っている cc 環境の UNIX のユーザ ID、パスワードを入力して接続ボタンを押します。パスワードの保存の所をチェックしておけば次に接続する時に今回入力したユーザ ID とパスワードを覚えていてくれます。パソコンを自分しか触らないのであればチェックしておいてもいいでしょう。多人数で使う可能性がある場合はチェックしてはいけません。



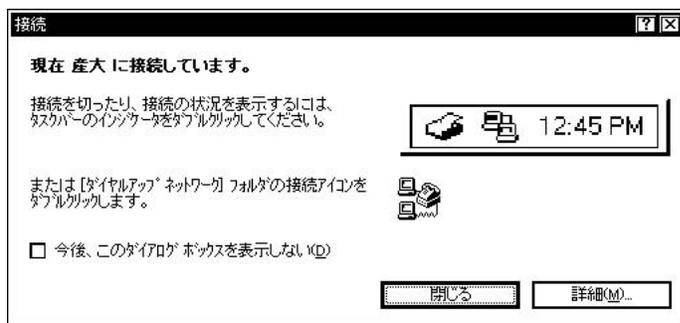
接続ボタンを押すと次のような画面に「ダイヤル中」「ユーザ ID とパスワードを確認中」「ネットワークにログオン中」「xxxbps で接続」と出ます。これで PPP 接続は成功です。後は Telnet や Netscape 等を起動してネットワーク接続がうまくいっているかどうか確認してください。Telnet などのソフトウェアのインストールについては Windows98 とほぼ同じであるため、Windows98 のほうにまとめました。そちらを参照してください。

もしうまく接続できないときは 341 ページの 11.9 を御一読ください。



## 補足

最近のパソコンではこのような画面がでるかもしれません。



この画面を閉じるとダイヤルアップ接続中を示すウィンドウがなくなります。これでは接続中なのか、切断されているのかわかりませんが、よく見るとタスクバーに二台のパソコンが線で結ばれているアイコンが表示されています。これがあると接続中、なければ切断されています。また、接続中ならこのアイコンで接続を切ることができます。



## 11.6 Windows98 編

以下に Windows98 が動作するコンピュータでダイヤルアップ IP 接続を行う例を示します。

### 11.6.1 用意する物

#### ハードウェア

- |                  |                         |
|------------------|-------------------------|
| PC/AT 互換機や PC-98 | Windows98 が動作していること。    |
| モデム              | 内蔵されているコンピュータもあります。     |
| モデムケーブル          | (大抵の場合はモデムを買うと付属しています。) |

#### 必要なソフトウェア

まずは接続のために。

Windows98 標準のダイヤルアップ機能。

接続してネットワークサービスを利用するために。

- |                   |                                      |
|-------------------|--------------------------------------|
| Telnet            | Windows98 に標準添付しているが Tera Term 等を推奨。 |
| Ftp               | Windows98 標準添付。                      |
| Internet Explorer | Windows98 標準添付。WWW ブラウザ。             |

#### あれば便利なソフトウェア

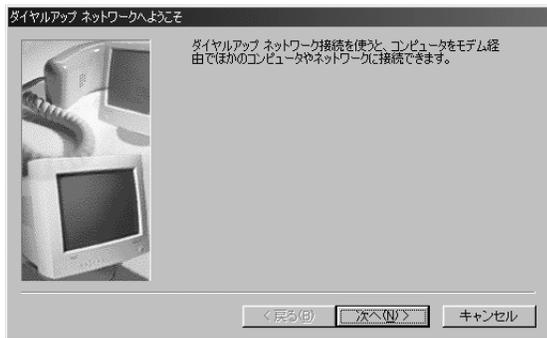
ネットワークサービスをより便利に利用するために。インストールの方法は説明しません。必要なら各ソフトに付属の説明書を読んでお使いください。

- |         |                            |
|---------|----------------------------|
| WS_FTP  | FTP ソフト。                   |
| AL-Mail | 電子メールソフト。最新の物を推奨。学生の利用は無料。 |
| WinVN   | ネットニュースソフト。最新の物を推奨。利用は無料。  |

### 11.6.2 インストール

マイコンピュータをダブルクリックします。開いたウィンドウから「ダイヤルアップネットワーク」を探してください。ダイヤルアップネットワークが無い人は Windows95 の説明を参考にダイヤルアップネットワークサービスを追加してください。「ダイヤルアップネットワーク」をダブルクリックして、「新しい接続」をダブルクリックします。ようこそ、という画面が表示されるので「次へ」を選びます。





ここでモデムが設定されていない場合は、モデムのインストールが始まります。モデムのインストールは機種ごとに異なるので、モデムの説明書を読みながらインストールしてください。

モデムのインストール途中に次のような設定画面が出てきます。これはコントロールパネルのモデムを選択し、モデムのプロパティを選択して設定することもできます。お使いの電話にあわせて、市外局番、外線発信番号、ダイヤル方法にそれぞれ入力します。外線発信番号は0発信なら0を入れます。外線を掛ける時に何もせずに直接掛けられるのならここは空白にしておきます。ダイヤル方法はトーンかパルスか<sup>19</sup>を選択します。



モデムのセットアップができれば、新しい接続として接続名を入力します。とりあえず京産大とでもしておきましょう。

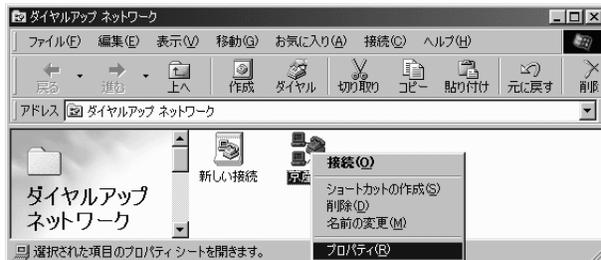
<sup>19</sup> 302 ページ、11.3.1 をご覧ください。



次を選ぶと接続先の電話番号を聞いてきますので、  
 市外局番 075  
 電話番号 705-3005  
 国番号 日本  
 として「次へ」を選び、「完了」を選びます。



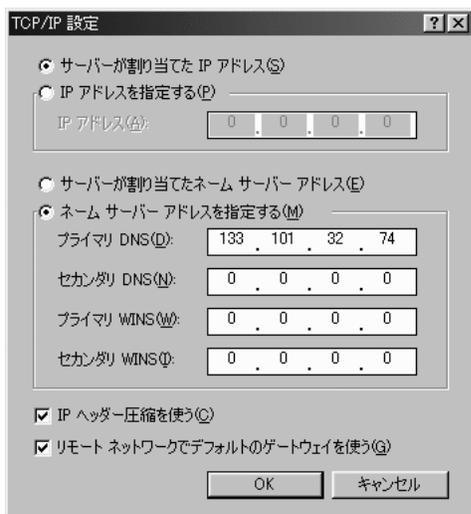
これでダイヤルアップネットワークの「新しい接続」横に「京産大」が作成されているでしょう。これを右クリックしてプロパティを選んでください。



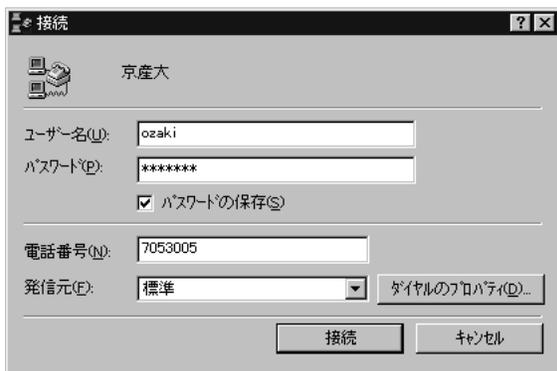
現れた画面から「サーバの種類」タグを選択し、図のようにチェックして「TCP/IP 設定」のボタンを押します。



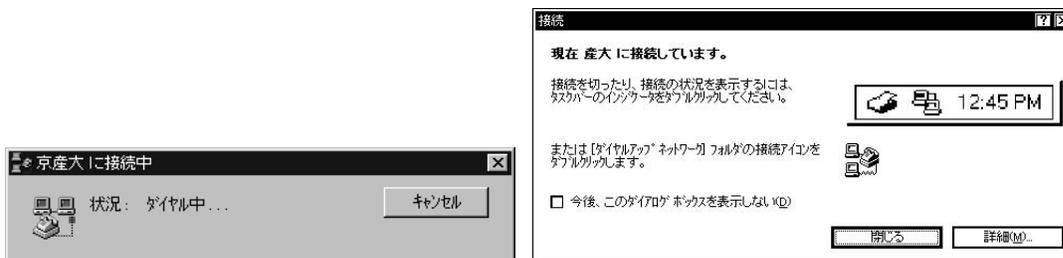
TCP/IP の設定を図のように「サーバーが割り当てた IP アドレス」、「ネームサーバーアドレスを指定」、「プライマリ DNS 133.101.32.74」と設定して OK を押します。他のウィンドウも OK を押して閉じてください。これで大学に接続するための設定は終了です。



では京産大のアイコンをダブルクリックしてみましょう。次のような画面が出てきますのでいつも学校で使っている cc 環境の UNIX のユーザ ID、パスワードを入力して接続ボタンを押します。パスワードの保存の所をチェックしておけば次に接続する時に今回入力したユーザ ID とパスワードを覚えていてくれます。パソコンを自分しか触らないのであればチェックしておいてもいいでしょう。多人数で使う可能性がある場合はチェックしてはいけません。



接続ボタンを押すと次のような画面に「ダイヤル中」「ユーザ ID とパスワードを確認中」「ネットワークにログオン中」の後、接続成功を示す画面が表示されます。これで PPP 接続は成功です。もしうまく接続できないときは 341 ページの 11.9 を御一読ください。



後は Telnet や Netscape 等を起動してネットワーク接続がうまくいっているかどうか確認してください。

#### 補足

接続成功の画面を閉じるとダイヤルアップ接続中を示すウインドウがなくなります。これでは接続中なのか、切断されているのかがわかりませんが、よく見るとタスクバーに二台のパソコンが線で結ばれているアイコンが表示されています。これがあると接続中、なければ切断されています。また、接続中ならこのアイコンで接続を切ることができます。



### 11.6.3 ネットワークソフトウェアの入手およびインストール

あなたの自宅からダイヤルアップ IP 接続の環境をより便利なものにするためにあると便利なアプリケーションのインストールを紹介します。以下にそって作業してください。ここではソフトウェアをネットワーク経由で手に入れる方法を取っています。これは時間がかかり、また電話代もかかります。雑誌で似たような作業の紹介とソフトウェアが入った CD-ROM が付いているものがあるのでそれを利用するのがお勧めです。また、最初から必要なソフトウェアがパソコンにインストールされているかもしれません。その場合はそちらを使えばいいですから、無駄な作業をしないように該当箇所を読み飛ばしてください。おおまかな作業は次のようになります。

- ダイヤルアップ IP 接続がつながった状態にする。
- Windows95・98 標準の FTP を利用して各種ソフトウェアを手に入れる。
- ダイヤルアップを一旦終了する。
- Lhasa をインストールする。
- Teraterm をインストールする。
- Netscape をインストールする。
- ダイヤルアップ IP 接続がつながった状態にして実際にソフトウェアを使う。

ここでインストールの説明をするソフトは以下のとおりです。なお、これらのソフトウェアはバージョンアップされることが予想されます。新しいものはなんらかの誤動作をしていたものが直っていると思われるので新しいものを使うことをお勧めします。その場合似た名前で、数字部分が大きな数字になったファイル名になっているものを選んでください。

種類	ソフト名	取得先 (平成 11 年 4 月 8 日現在)
Telnet	Teraterm	ftp://ftp.kyoto-su.ac.jp/pub/Windows/NetTools/Telnet/teraterm/tterm23.zip
WWW	Netscape	ftp://ftp.kyoto-su.ac.jp/pub/Windows/NetTools/WWW/Netscape/english/n32e408.exe
Archiver	Lhasa	ftp://ftp.kyoto-su.ac.jp/pub/Windows/DiskTools/lhasa011.exe

さて、上記の表記で判る方は好きな方法でファイルを取得してください。Windows98 など持っているパソコンに最初から Netscape 等の WWW ブラウザがインストールされているなら、それを起動して上記の URL にアクセスするとパソコンのハードディスクに保存できるでしょう。判らない方は、下記を参考にチャレンジしてください。

#### 各種ソフトウェアの取得

ダイヤルアップ IP 接続がつながった状態で「スタート」メニューから「プログラム」を選び、「MSDOS プロンプト」を起動します。続いて以下のように入力してください。下線部が入力する部分です。

```
C: ¥ WINDOVS>cd ¥
```

```
C: ¥ >ftp ftp.kyoto-su.ac.jp
```

するとユーザ ID を聞いてきます。ここで入力するユーザ ID は cc 環境のユーザ ID とは異なります。ユーザ ID に anonymous と入力してください。続いてパスワードを聞いてきますが、ここには自分の電子メールアドレス (例: hoguehoge@cc.kyoto-su.ac.jp) を入力します。

```

C: ¥>ftp ftp.kyoto-su.ac.jp
Connected to ccnic07.kyoto-su.ac.jp.
220 ccnic07 FTP server (Version wu-2.4.2-academ[BETA-15](1) Sat Jan 10 19:22:30
JST 1998) ready.
User (ccnic07.kyoto-su.ac.jp:(none)): anonymous
331 Guest login ok, send your complete e-mail address as password.
Password: hogehoge@cc.kyoto-su.ac.jp ( 画面には表示されません )
230-#####
230-# Welcome to the Kyoto Sangyo University Computer Center archive.
230-# All archive files are stored under /pub.
230-# Please report if you have problem to ozaki@cc.kyoto-su.ac.jp
230-#####
230-
230 Guest login ok, access restrictions apply.
ftp>

```

きちんと入力できるとこのように数行のメッセージの後、「Guest login ok」と出て入力待ちになります<sup>20</sup>。失敗して login failed と表示されたなら anonymous のスペルを入力ミスしていると思われます。quit と入力して再度チャレンジしてください。

無事例のようになったら続けて次のようにします。それぞれの入力後に例と違うメッセージが出たら入力間違いをしていると思われます。入力したコマンドをよく確認して、間違っていたら再入力してください。間違っていないのにおかしなメッセージが出る時は何か変更があったのかもしれませんが。ネットニュースの sandai.question など質問してみましょう。

```

ftp> bin
200 Type set to I.
ftp> cd /pub/Windows/NetTools/teraterm
250 CWD command successful.
ftp> get tterm23.zip
200 PORT command successful.
150 Opening BINARY mode data connection for tterm23.zip (943376 bytes).
226 Transfer complete.
943376 bytes received in 262.04 seconds (3.60 Kbytes/sec)
ftp>

```

get tterm23.zip を実行してから complete と出るまではかなり時間がかかります。モデムの速度によりますが 28800bps のモデムで 5 分近くかかるでしょう。ここまでで tterm23.zip が自分のパソコンのハードディスクにコピーされました。

続いて同様に lhasa011.exe を入手します。これは 1 分ほどでしょう。

```

ftp> cd /pub/Windows/DiskTools
250 CWD command successful.

```

<sup>20</sup> PC-98 シリーズの場合、この作業がうまくいかない (途中で止まってしまう) 場合があります。その時は回避策がありませんので雑誌の付録 CD-ROM からか学校の情報処理教室でフロッピーディスクを利用してファイルを取得するなど、他の方法を試みてください。

```
ftp> get lhasa011.exe
200 PORT command successful.
150 Opening BINARY mode data connection for lhasa011.exe (184320 bytes).
226 Transfer complete.
184320 bytes received in 51.20 seconds (3.60 Kbytes/sec)
ftp>
```

さらに必要なら Netscape を入手できますが、これには 40 分近く時間がかかります。可能なら別の方法で手に入れることをお勧めします。Windows98 なら標準で Internet Explorer がインストールされていますからそれを使えばいいでしょう。

```
ftp> cd /pub/Windows/NetTools/WWW/Netscape/english
250 CWD command successful.
ftp> get n32e408.exe
200 PORT command successful.
150 Opening BINARY mode data connection for n32e408.exe (10207782 bytes).
226 Transfer complete.
10207782 bytes received in 2647 seconds (3.60 Kbytes/sec)
ftp>
```

必要なソフトウェアを手に入れたら quit とコマンドを入力してください。そして一旦ダイヤルアップ接続を切ります。

## Lhasa のインストール

### 予備知識 - 圧縮と解凍 -

コンピュータは非常にたくさんのファイルを扱っていることは既におわかりのことと思います。それらは友人からきた電子メールがもしれませんし、ワープロソフトで作った文書ファイルがもしれません。さて、このようなファイルは得てしてどんどん増え続けるものです。そうすると置き場所がなくなって困ってしまいます。どうすればいいのでしょうか？ここで活躍するのが圧縮です。

ふとんの真空パックを想像してください。大きなサイズのふとんがぎゅっと圧縮されて小さくなりますね。すると、そのままの大きさでは 5 枚しか入らない押し入れに 10 枚押し込むことができるがもしれません。欠点はそのまま使うことができず、空気を入れて元の大きさに戻さないと使えないことです。しかしその欠点を理解した上で非常に便利に利用することができます。

このようなことがコンピュータではファイルに対して実行できるのです。圧縮して小さくして<sup>21</sup> 保存しておき、必要ときに元に戻す作業、解凍をして使うのです。さて、これだけでは芸がありません。圧縮とセットで使うことで格段に整理整頓が便利にできる機能があります。いくつかのファイルをまとめて一つにしてしまう機能です。これをコンピュータの世界ではアーカイブ (archives<sup>22</sup>) と呼んでいます。アーカイブしたファイルを圧縮すれば、あるまと

<sup>21</sup> 例えば、「aaaaabbbcccc」というデータを a が 5 個、b が 3 個、c が 4 個とみて「a5b3c4」とするのです。12 文字あった元のデータが 6 文字と、半分の大きさになったことが判りますか？

<sup>22</sup> archives 本来の意味は公文書、とか公文書保管所です。まとめて片づけるという点では共通していますね。

まったデータを片づけておくことができます。圧縮・解凍をするソフトウェアは普通アーカイブ機能も備えています。そのため圧縮・解凍ソフトウェアをアーカイバと呼ぶこともあります。

この機能は個人のファイルの整理にももちろん利用できますが、それよりも利用価値が高いのが、さまざまなソフトウェアを提供する場合です。ソフトウェアをアーカイブ・圧縮された一つのファイルの形にしておけば相手に過不足なくすべてのファイルを渡すことができますし、手軽に、通信を通じてなら短時間で、フロッピーディスクなら少ない枚数で渡すことができます。世の中にはこのような形でたくさんのソフトウェアが提供されています。

さて、tterm23.zip というファイルを取得しましたが、このファイルは Tera Term Pro を圧縮したファイルなので、使えるように解凍しなくてははいけません。拡張子からこのファイルは ZIP 形式で圧縮されていることがわかりますので、ZIP 形式の解凍ソフトが必要になります。これが一緒に取得した Lhasa です。

Windows95 の「スタートメニュー」から「ファイル名を指定して実行」を選んで C:\¥lhasa011.exe と入力し、OK を押してください。インストールする場所を選ぶよう指示があります。そのまま OK でいいでしょう。

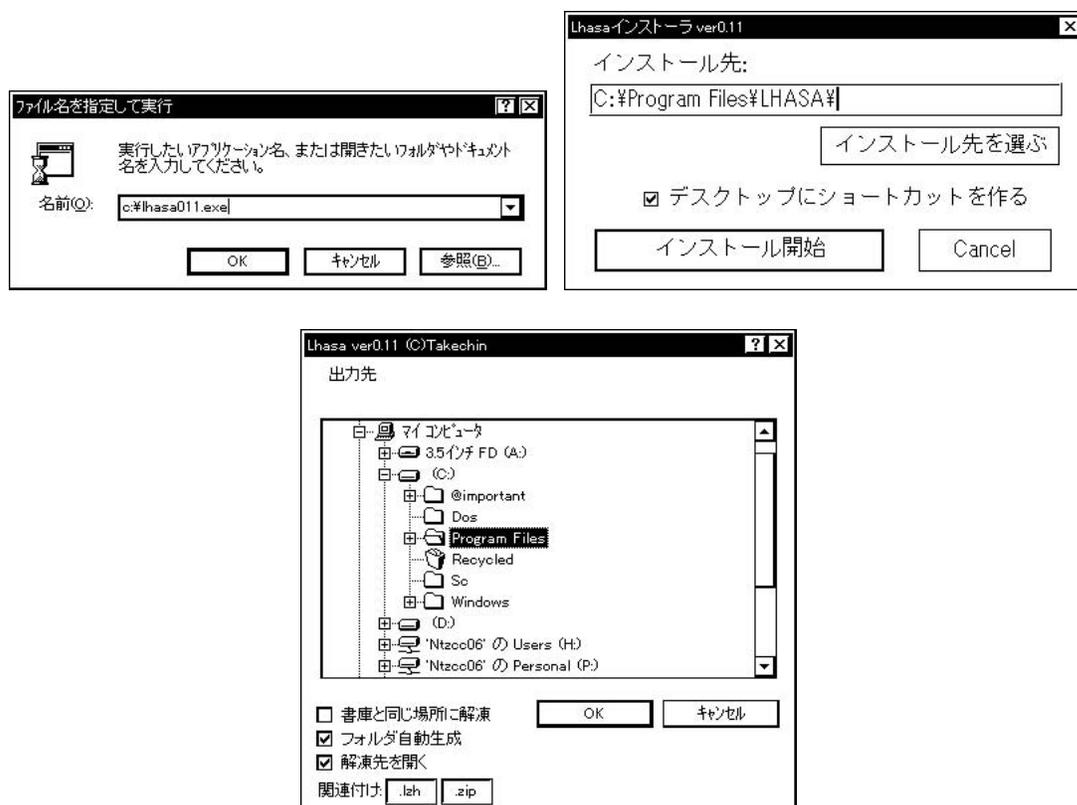


図 11.3 Lhasa インストール画面

「インストール完了」というボタンを押すと設定ウィンドウが開きます。図のように「フォルダ自動生成」「解凍先を開く」「関連付け lzh, zip」を選択して、出力先を C ドライブの Program Files に指定して OK ボタンを押してください。

## Tera Term Pro のインストールと設定

Windows95 のデスクトップにあるマイコンピュータをダブルクリックしてください。現れたウインドウから、(C:) をダブルクリック、更に現れたウインドウから ttermp23.zip<sup>23</sup> をダブルクリックしてください<sup>24</sup>。棒グラフがかかれたようなウインドウが現れた後、同じようなウインドウがまた開きます。今度は Setup.exe<sup>25</sup> をダブルクリックしてください。自動的に Tera Term Pro のインストールが始まります。画面に従って進めてください。

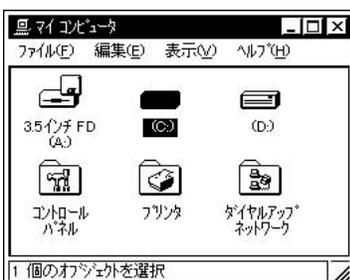


図 11.4 マイコンピュータダブルクリック時に現れるウインドウ

インストールが終了したら Tera Term Pro とタイトルのついたウインドウが開き、スタートメニューのプログラムに Tera Term Pro が追加されます。Readme を開いて目を通してください。

Readme に目を通したら Tera Term Pro を起動してください。最初の画面はとりあえずキャンセルしてください。

Setup メニューから Terminal を選び、Kanji(receive) と Kanji(transmit) の項目を EUC に変更します。これをしないと cc2000 に接続したときに日本語が表示されません。他の項目は特に変更する必要はないでしょう。OK ボタンを押します。

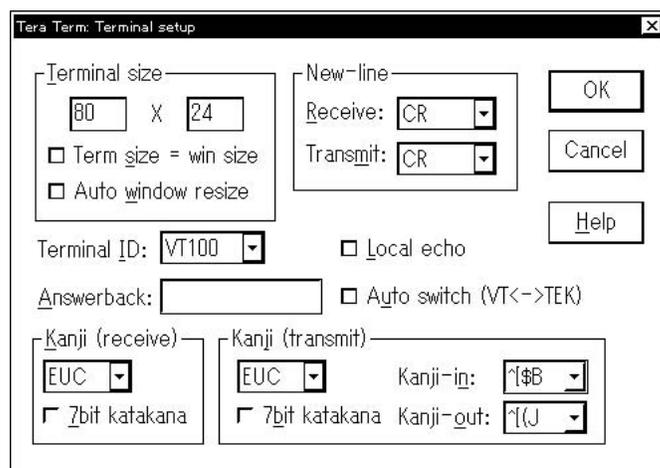


図 11.5 Tera Term Pro 設定画面

<sup>23</sup> ttermp23 としか書かれていないかもしれません。

<sup>24</sup> ここで「ファイル'ttermp23.zip'を開くアプリケーションを選んでください」と書かれたウインドウが開くようなら Lhasa の設定が正しく行えていません。キャンセルして Lhasa の設定を再度行ってください。

<sup>25</sup> Setup としか書かれていないかもしれません。

次は Setup メニューから Save setup... を選び、そのまま <return>を押してください。これをしないと毎回上記の日本語設定をしなければなりません。

以上で Tera Term PRO の設定は終わりです。実際に利用するときはダイヤルアップ IP 接続時に起動して、最初の画面で Host 欄に cc2000.kyoto-su.ac.jp と入力して OK ボタンを押せば cc2000 につながります。

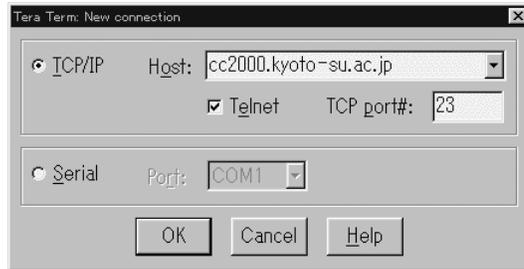


図 11.6 Tera Term Pro 起動画面

## Netscape のインストール

エクスプローラーから n32e408.exe をダブルクリックすると自動的にインストーラが起動するので画面に従って進めてください。「次へ」でどんどん進んで「セットアップを続ける」という選択をし、ライセンスに同意すると Netscape が起動します。

## Netscape 等、WWW ブラウザの設定

Netscape に限らず、Microsoft インターネットエクスプローラ (以下 MSIE) など、各種の WWW ブラウザには以下の項目を設定してください。

- 文字コードの設定 (Netscape のみ。)

Netscape 3.0 以前は Setup メニューから Documet Encoding、Netscape 4.0 なら View メニューから Encoding を選択し、Japanese (Auto-Detect) に設定します。これをしないと日本語が正しく表示されません。設定後、set default をしないと設定が保存されません。

- Proxy (プロキシ) サーバの設定

Proxy とは本来セキュリティ対策に関連した、ネットワーク中継をするための機能なのですが、WWW の Proxy サーバにはもう一つの役割が付加されていて、最近では Proxy というとその付加された役割を指すことが多いようです。付加された役割とは専門的にいうと WWW データのキャッシング、わかりやすくいうと一度 WWW で見たデータを覚えておいて、もう一度見ようとしたときに高速に表示する機能です。

もう少し具体的に説明します。A という WWW の Proxy サーバがあるとします。通常 WWW ブラウザでいろいろなページを見ていると、その都度ネットワークを介して遠く離れた所にあるコンピュータまで情報を取りに行きます。それに対して WWW ブラウザの Proxy サーバに A を設定しているようなページを見たときは、まず A にその情報がないかどうかを問い合わせます。A がその情報を持っていたらすぐにその情報を返してくれます。わざわざ遠くにあるコンピュータまで情報を取りに行かずに近くの A から情報を取ることで高速に情報が送られてきます。A が情報を持っていなかった

時はブラウザの代わりに A がそのデータを取りに行き、データを取った後、ブラウザに情報を送ると共に A はその情報を覚えます。これで次にもう一度その情報を参照した時は遠くのコンピュータではなく、A が情報を覚えているので A から情報が送られてきます<sup>26</sup>。

さて、Proxy サーバの真価は大勢が利用した時に発揮されます。大勢の人が使っていても、それぞれがばらばらの WWW ページを参照しているわけではありません。むしろ大多数は人気の高い同じ WWW ページを見ているでしょう。Proxy サーバがないとそれぞれが遠くのコンピュータから情報を取り寄せてしまいます。その結果学外のネットワークが混み合い、全体的に対外的な通信速度が低下します。しかし Proxy サーバがあると、同じページを見ている人のデータ転送は WWW ブラウザと Proxy サーバの間でしか行なわれません。その結果同じページを見ている人の表示速度は速くなり、そうでない人も学外のネットワークの渋滞が緩和されるため順調に情報を取得することができるのです。

いかがでしょう。Proxy サーバの意義が判って頂けましたか？判って頂けたのなら次のように設定してください。但しこの設定は京都産業大学に電話で接続しているときだけです。一般のプロバイダなどに接続しているときは利用できません<sup>27</sup>。

Proxy サーバ名   wwwproxy.kyoto-su.ac.jp  
ポート番号       8000

主なブラウザでの設定画面を表示する手順は次のとおりです。

- Netscape3.0 以前 Option(オプション) メニューから Network Preferences(ネットワークの設定) を選択。Proxies(プロキシ) タグを選び Manual Proxy Configuration(手動) にして、View(表示) ボタンを押します。
- Netscape4.0 Edit(編集) メニューから Preferences(設定) を選ぶとウインドウが開きます。カテゴリの枠から Advanced(詳細) の Proxies(プロキシ) を選ぶと右側に Proxies の設定画面が出てきますので Manual proxy configuration(手動でプロキシを設定する) の左にチェックして、更にその右側の View(表示) ボタンを押します。
- MSIE4.0 表示メニューからインターネットオプションを選ぶとウインドウが出てきます。接続タグを選び、「プロキシサーバを使用してインターネットにアクセス」の左側のボックスにチェックを入れ、右側の詳細ボタンを押します。

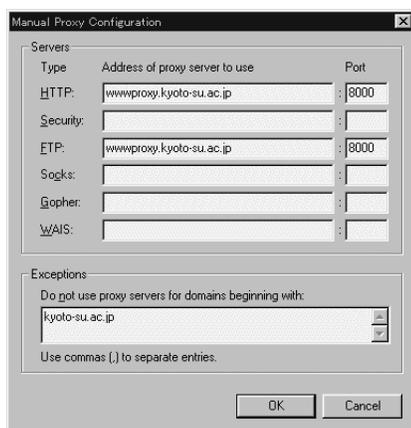


図 11.7 Netscape Proxy 設定



IE4.0 Proxy 設定

### ● ホームページの設定

ブラウザは標準のインストールでは起動時に Netscape 社や Microsoft 社につながるように設定してあります。必要に応じて産大のホームページを設定してください。

<sup>26</sup> このように遅いメディアのデータを速いメディアに記憶させて速度を向上させる技術をキャッシングと呼びます。WWW データに限った技術ではないのでここでは WWW データのキャッシングと呼ぶ方がいいかもかもしれません。

<sup>27</sup> きっとそのプロバイダにも Proxy サーバがあるのでそれを使ってください。

## 11.7 Macintosh 編

以下に Macintosh (漢字 Talk7.5 以降が動作しているもの) を用いてダイヤルアップ IP 接続を行う例を示します。

### 11.7.1 用意するもの

#### ハードウェア

Macintosh	漢字 Talk7.5 以降が動作していること。
モデム	本体購入時に内蔵されてくる場合もある。 Open Transport/PPP(以降 OT/PPP と記述) を利用する場合は、OT/PPP に 対応したモデムを用意すること。
モデムケーブル	Macintosh 用としてモデムを購入すると付属している場合もある。 モデムが本体に内蔵されている場合は必要ない。

#### 必要なソフトウェア

##### まずは接続のために

MacTCP または TCP/IP	システム CD-ROM に格納されています。TCP/IP は Open Transport をインストールすると同時にインストールされます。
OT/PPP または MacPPP 等	OT/PPP はシステム CD-ROM に格納されています。 MacPPP を利用する場合はバージョン 2.2.0a 以降を推奨。 (注意!) 一部の機種では、特定のバージョンの MacPPP がうまく動作しない場合があります。このような場合は MacPPP のバージョンを変えて試してみたり、FreePPP <バージョン 2.5v2 以降を推奨> 等他のダイヤルアップ IP 接続ソフトを試してみてください (FreePPP 等他のダイヤルアップ IP 接続ソフトを利用する場合は、付属の説明書及び以降の説明を参照しながら設定を行ってください)。それでもだめならメーカーのユーザーサポートセンターにご相談ください。

MacPPP、FreePPP はネットワーク上で流通しているので入手は容易です。Internet や Macintosh 関係雑誌の附録 CD-ROM に含まれていることも多いでしょう。

##### 接続してネットワークサービスを利用するために

以下のソフトウェアについては自分が必要とするものだけをそろえればよいでしょう。

Telnet 端末エミュレータ	Better Telnet など。
ftp クライアント	Fetch など。
WWW ブラウザ	Netscape Navigator など。
電子メールクライアント	Eudora など。

これらはネットワーク経由で大抵取得できます。他にも様々なアプリケーションが利用できますが、ここではこれ以上挙げません。また、このようにネットワークなどを用いて配布されているソフトウェアには、無料のもの(条件付きで無料のものもある)、有料のものが存在します。詳しくは、各ソフトウェア添付の書類を確認してください。

## 11.7.2 インストール

<重要!! 選択する作業パターンを確認する>

利用する Macintosh のオペレーティングシステムのバージョンによって、行うべき作業は次の3パターンに分けられます。各パターンを選択した場合に、それぞれ作業せねばならぬ項目をあげておきますので、以降はこれをガイドに設定を進めて行ってください。

パターン1) 漢字 Talk7.5.x 利用者

旧ネットワークソフト (MacTCP) を利用します。一部のバージョンを除き、ダイヤルアップ IP 接続ソフトがシステム CD-ROM にはっていないため、自分で入手しなくてはなりません。なお 7.5.2 以降のバージョンでは Open Transport が利用できる場合もありますので、詳しくは添付される説明書を参照してください。

- MacTCP のインストール
- MacPPP のインストール
- MacTCP の設定
- MacPPP の設定
- 接続する (MacPPP 利用時)

パターン2) MacOS7.6.x または MacOS8.x 利用者 (OT/PPP を利用する場合)

Open Transport (TCP/IP) を利用します。純正ダイヤルアップ IP 接続ソフトである Open Transport/PPP (以降、OT/PPP と記述) が、システムソフトウェアの入った CD-ROM (以降、システム CD-ROM と記述) に格納されています。しかしながら OT/PPP に対応しているモデムが必要です。

- Open Transport(TCP/IP) のインストール
- OT/PPP のインストール
- TCP/IP の設定
- OT/PPP (及びモデム) の設定
- 接続する (OT/PPP 利用時)

パターン3) MacOS7.6.x または MacOS8.x 利用者 (OT/PPP を利用しない場合)

MacPPP を使います。OT/PPP に対応していないモデムでパターン2が選択できなかった場合です。

- Open Transport (TCP/IP) のインストール
- MacPPP のインストール
- TCP/IP の設定
- MacPPP の設定
- 接続する (MacPPP 利用時)
- 接続を切る (MacPPP 利用時)

## MacTCP または OpenTransport(TCP/IP) のインストール

- MacTCP を利用する場合

すでにインストールされている場合は、コントロールパネルフォルダ内に「MacTCP」コントロールパネル () が存在しているはずですが、まだインストールされていない場合は、システム CD-ROM を用いてインストールしてください。詳しくは本体付属のマニュアルを参照してください。

- Open Transport を利用する場合

すでにインストールされている場合は、コントロールパネルフォルダ内に「TCP/IP」コントロールパネル () が存在しているはずですが、まだインストールされていない場合は、システム CD-ROM からインストールしてください。詳しくは本体付属のマニュアルを参照してください。

< Open Transport をインストールしたはずなのに、コントロールパネルフォルダ内に「TCP/IP」コントロールパネルが見当たらない！というひとへ>この状態が発生するのは漢字 Talk7.5.2～7.5.5 で Open Transport を利用しようとしている場合に限られるでしょう。そのかわりに「MacTCP」コントロールパネルがありませんか？もしそうなら「ネットワークソフト選択」が行われていません。説明書をよく読んで切り替えてください。

## MacPPP のインストール

Config PPP アイコン () と PPP アイコン () をシステムフォルダの中にドラッグ&ドロップすれば「必要なフォルダ以下に入れるか？」と聞いてくるので「はい」と答える。要はコントロールパネルフォルダ以下に、ConfigPPP を、機能拡張フォルダに PPP をコピーすれば構いません。

なお、Config PPP アイコンは MacPPP のバージョンによっては「Config PPP-J」という名前になっている場合があります。

## OT/PPP のインストール

すでにインストールされている場合は、コントロールパネルフォルダ内に「PPP」() 及び「モデム」コントロールパネル () が存在しているはずですが、まだインストールされていない場合はシステム CD-ROM からインストールしてください。

< OT/PPP をインストールしようとしたら、「Open Transport のバージョンが古い」といわれインストールできない！>この状態が発生するのは漢字 Talk7.5.2～7.5.5 で Open Transport を利用しようとしている場合に限られるでしょう。これらのシステム CD-ROM に格納されている Open Transport では OT/PPP は利用できませんので、MacPPP 等他の PPP 接続ソフトを利用してください。また、Open Transport を諦めて MacTCP を利用するのも一方法でしょう（ネットワークを通じて Open Transport の新しいバージョンを手に入れることもできますが、お勧めはしません）。

## MacTCP または TCP/IP の設定

MacTCP を利用する場合は、コントロールパネルで MacTCP を選択すれば以下のような表示になります。

Open Transport を利用する場合は、コントロールパネルで TCP/IP を選択すれば図 11.9 の下図になります。もし同じような画面にならない場合は、「TCP/IP」コントロールパネルが選択されている状態で、

メニューバーの「編集」から「利用者モード」を選択し、「詳しい情報も指定」をクリックしてチェックをつけた後、「OK」ボタンをクリックしてください。これで表示内容が変更されているはずですが。

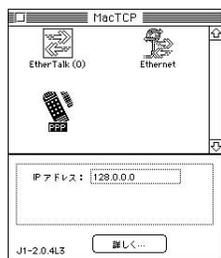


図 11.8 MacTCP で PPP を選択する

ここで MacTCP を利用する場合は、PPP アイコン (  ) を選択します。更に詳細を設定するために「詳しく...」ボタンをクリックすると以下のような表示になります。

Open Transport を利用する場合は、「経路先:」と書いてある右横のボックスを押し、MacPPP を選択します (OT/PPP 利用者は、PPP を選択してください)。次に各項目を設定しますが、図 11.9 を参照して同じ内容にしてください。

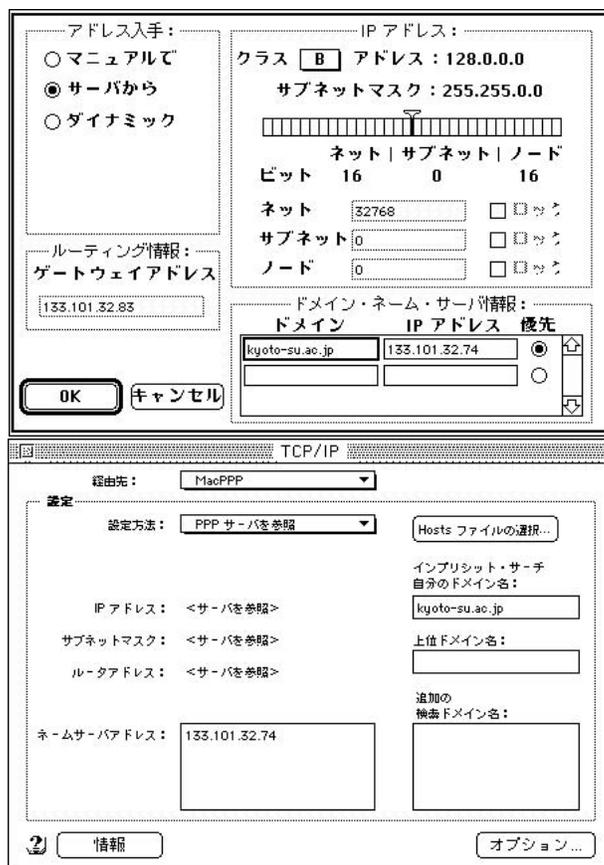


図 11.9 上: MacTCP 詳細設定、下: TCP/IP の設定 (Open Transport 利用)

各部の情報は上記と全く同じにするのが良いでしょう。ただし、IP アドレス情報のネット、サブネット、ノード情報は設定する必要はありません。

MacTCP を利用している場合、設定が終れば「OK」ボタンをクリックします。そして先に「詳細…」ボタンを押した MacTCP ウィンドウを閉じます。閉じた際に「MacTCP の設定を有効にしたければ再起動させよ」という意味のメッセージが現れるかもしれません。その際は Macintosh を再起動してください。

Open Transport を利用している場合は、TCP/IP コントロールパネルを閉じてください。すると設定を保存するかどうかを問い合わせてきますので”保存”を選択してください。

## MacPPP の設定

コントロールパネルで Config PPP を選択すれば以下のような表示になります。

なお、MacPPP のバージョンによっては、ウィンドウ内の表示が英語で行われる場合がありますが、同じように設定していけば問題ありません。



図 11.10 MacPPP 設定

各項目を上記の例に合わせて設定してください。

- ポート名はモデムポート (Modem Port)、プリンタポート (Printer Port) 等モデムが接続されているところを選んでください。なお最初は「Modem Port」と表示されていることがありますが、一旦メニュー選択をすると「モデムポート」などとカナ表示に変化します。
- PowerBook などの一部機種では「モデム / プリンタポート」などと表示される場合があります。これはモデムポートとプリンタポートを一つのポートで兼用している場合に表示されるのですが、この場合はアップルメニュー以下のセレクトラを利用して AppleTalk を「不使用」に設定しておく必要があります。
- 動作可能時間はどれだけの時間回線を使わなかったら自動的に切断するかを分で指定します。ただし、技術的な理由からこの設定は行わないことをお勧めします。

PPP Server をまだ一つも登録していませんから、ここで京都産業大学のダイヤルアップ IP サーバを登録します。「新規…」ボタンをクリックすれば以下のような表示になります。

図 11.11 サーバ名の登録

ここで「K.S.U」とサーバ名をタイプして「OK」ボタンをクリックします。すると「Config PPP」コントロールパネルの「PPP サーバ：」右横にいま登録したサーバ名が表示されているはずです。この状態で、設定... ボタンをクリックすると、以下のような表示になるでしょう。

図 11.12 サーバ詳細設定

各項目の情報は上記と全く同じにすれば良いでしょう。ただし、以下の項目は各自の状況によって変化することに注意してください。

- ポート速度については各自が利用しているモデムと Macintosh の間で安定して通信できる最高の速度にするのが良い。単位は bps。
- トーン か パルス かは各自の電話の契約に依存することに注意（設定を誤ることが多い項目のひとつです）。302 ページの 11.3.1 を参照して正しく設定してください。
- 電話番号については市外局番が必要になったり、局内交換機からゼロ発信するためのゼロが一つ余計に必要になったりすることに注意。

更に認証情報の設定をします。「認証...」ボタンをクリックすると以下のような表示になります。

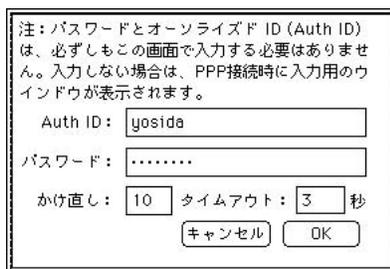


図 11.13 ユーザ ID とパスワードの登録

上記のようにあなたの cc 環境での UNIX のユーザ ID、パスワードを設定して「OK」ボタンをクリックしてください。なお、図にあるとおり必ずしもここで オーソライズド ID とパスワードをいれる必要はありません（その場合ダイヤルアップ IP 接続時にパスワード入力を要求する画面が表示されます）が、もし入力枠の中に、入れた覚えがない文字などが表示されている場合は、きれいに消去しておかないと接続時トラブルの原因となります。

これで MacPPP の設定は終了です。接続用スクリプト、LCP オプション、IPCP オプションについては特に必要な場合を除いて設定する必要はありません。

#### OT/PPP（及びモデム）の設定

コントロールパネルでモデムを選択すれば以下のような表示になります。



図 11.14 モデムの設定 (OT/PPP)

各項目を上記の例に合わせて設定してください。

- 経路先：モデムポート、プリンタポート等、モデムが接続されているところを選んでください。
- モデム：使用するモデムの機種を選択してください。この一覧にない場合はモデムについてきた説明書を読んで対処してください。それでもわからない場合は、メーカーに確認してください。
- スピーカー：接続が確認できるまでとりあえず「入」を選んでおいてください。
- ダイヤル：設定を誤ることが多い項目のひとつです。302 ページの 11.3.1 を参照して正しく設定してください。

- ダイアルトーンを無視：受話器をあげたとき、「ツー」という連続音が聞こえないような電話回線を利用する場合、この項目をチェックする必要があるかもしれません。

以上設定が終わったなら、モデムコントロールパネルを閉じてください。設定を保存するかどうか聞いてきますので、保存を選択してください。

次に、コントロールパネルから PPP を選択すれば以下のような設定になります。

図 11.15 PPP の設定 (OT/PPP)

各項目を上記の例に合わせて設定してください。

- ユーザー ID、パスワード：cc 環境の UNIX でのあなたのユーザー ID、パスワードを設定してください。
- パスワード保存：例ではチェックされていますが、必ずしもチェックする必要はありません（チェックしない場合、ダイヤルアップ IP 接続時にパスワード入力を要求する画面が表示されます）。
- 電話番号：市外局番が必要になったり、局内交換機からゼロ発信するためにゼロが一つ余計に必要になったりすることに注意。

設定が終わったなら、PPP コントロールパネルを閉じてください。設定を保存するかどうか聞いてきますので、保存を選択してください。

### 11.7.3 使い方

#### 接続する (MacPPP 利用時)

コントロールパネルで Config PPP を選択すれば以下のような表示になります。



図 11.16 MacPPP

ここで PPP サーバとして先程登録した「K.S.U」などが選択されていることを確認してください。それが済めば「開く」ボタンをクリックします。

(注意!) この段階で「開く」ボタンが淡色表示されており、クリックしても反転表示されない場合は、MacTCP または Open transport の設定を再確認してください。もしくは作業途中で、画面に「設定を有効にするため再起動してほしい」旨メッセージがでていたが、まだ再起動していないということはないでしょうか。

モデムが正しく電話回線に接続されていれば、ここで以下のようなウインドウが表示され電話のダイヤル音が聞こえてくるでしょう。



図 11.17 ダイヤル中の表示

そしてネットワーク接続が完了すると、このウインドウは自動的に閉じます(ここまでにおおよそ数十秒を要します)。これでダイヤルアップ IP 接続は済みました。なお、以下のような確認の表示が出るかもしれませんので、OK ボタンをクリックしてください。



図 11.18 接続成功の表示

後は実際に Telnet 端末エミュレータや WWW ブラウザなどを起動して、ネットワーク接続がうまく行っ

ていることを確認してください。Config PPP コントロール・パネルのウインドウは閉じて構いません（接続していることを忘れないように）。

因みにこの時、Config PPP コントロールパネルのウインドウの左上に「PPP UP」と書かれたアイコンが表示されているでしょう。

#### 接続を切る（MacPPP 利用時）

Telnet 端末エミュレータや WWW ブラウザなど、ネットワークアプリケーションが終了していることを確認してください。全て終了していることを確認後、コントロールパネルで Config PPP を選択すれば接続成功時と同様の表示になります。

ここで Config PPP コントロールパネル内の「閉じる」ボタンをクリックします。しばらくすると電話回線が切れます。これで完了です。

因みにこの時、Config PPP コントロールパネルのウインドウの左上には「PPP DOWN」と書かれたアイコンが表示されているでしょう。

#### 接続する（OT/PPP 利用時）

コントロールパネルで PPP を選択すれば以下のような表示になります。

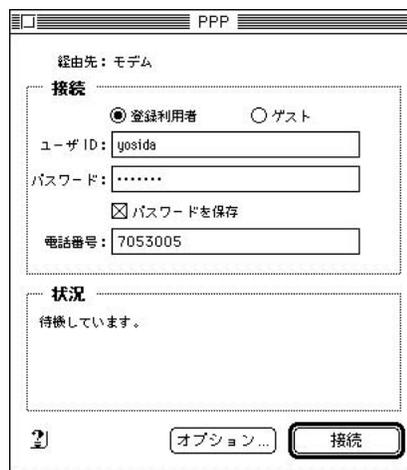


図 11.19 OT/PPP

接続（およびモデム電源が入っているか等）を確認し、大丈夫なら「接続」ボタンをクリックしてください。

なにか問題があればその内容が表示されますので、その情報を手がかりに対処してください。

モデムが正しく接続されていれば、PPP コントロールパネルの「状況」ウインドウに接続中である旨が表示され、ほどなく電話のダイヤル音が聞こえてくるでしょう。

そののち、「状況」ウインドウに次々と状況が表示され、接続が完了すると、次のような表示となるでしょう。



図 11.20 接続成功の表示 (OT/PPP)

後は、実際に Telnet 端末エミュレータや WWW ブラウザなどを起動して、ネットワーク接続がうまくいっていることを確認してください。PPP コントロールパネルは閉じて構いません（接続していることを忘れないように）。

#### 接続を切る (OT/PPP 利用時)

Telnet 端末エミュレータや WWW ブラウザなど、ネットワークアプリケーションが終了していることを確認してください。全て終了していることを確認後、コントロールパネルで PPP を選択すれば接続成功時と同様の表示になります。

ここで、「解除」ボタンをクリックします。「状況」ウインドウに接続を解除中である旨表示され、しばらくすると電話回線が切れます。これで完了です。このとき「状況」ウインドウには、モデムが待機中である旨表示されているでしょう。

## 11.8 慣れている人のために

以下に電話線経由でのコンピュータの接続に慣れている人のために、必要となる情報を示しておきます。

京都産業大学計算機センターが現在用意している設備には以下のものがあります。しかし各設備は随時構成変更の可能性があります。変更があった時点で sandai.center.announce などの学内ニュースグループなどに掲示されますのでチェックしておくといいいでしょう。

### 11.8.1 ダイヤルアップ IP 接続に関する情報

PPP によるダイヤルアップ IP 接続の為の設備と、その設定情報を以下に示します。

#### 受信用電話回線とモデム

##### (タイプ I:CISCO 社 AS5200)

接続速度が高速 (アナログ 56Kbps, デジタル (ISDN)64Kbps) で、回線数を多く (46 つ) とっていますので通常はこちらを御利用ください。

---

回線数	46
電話番号	075-705-3005 (内線番号はありません)
製品名	AS5200 内臓モデム
通信速度	64K(ISDN), 56K(k56flex), 38400, 28800, 26400, 24000, 19200, 16800, 14400, 12000, 9600, 7200, 4800, 2400, 1200, 300bps

##### (タイプ II:Livingstone 社 PortMaster)

接続速度が比較的高速 (28.8Kbps) で、回線数を多く (8 つ) とっています。タイプ I が使えないときに御利用ください。

---

回線数	8
電話番号	075-705-1502 (内線 2992)
製品名	Microcore 社製 MC288XL
通信速度	28800, 26400, 24000, 19200, 16800, 14400, 12000, 9600, 7200, 4800, 2400, 1200, 300bps

##### (タイプ III:Livingstone 社 PortMaster)

接続速度が低速 (14.4Kbps) で、回線数も少なく (5 つ) になっており、通常は他のタイプを利用する方が有利です。他のタイプにつながらない時に御利用ください。

---

回線数	5
電話番号	075-705-1489 (内線 2592)
製品名	OMRON 社製 MD144XT10V
通信速度	14400, 12000, 9600, 7200, 4800, 2400, 1200, 300bps

## 設定

接続するためのソフトウェアの設定は以下のようにしてください。

---

ユーザ ID	cc 環境の UNIX のユーザ ID
パスワード	cc 環境の UNIX のパスワード
認証方式	Connect Script 及び PAP。CHAP (暗号化パスワード) には対応していません。
IP アドレス	サーバから割り当て
ネットマスク	255.255.0.0
ドメインネームサーバ	133.101.32.74
ドメイン名	kyoto-su.ac.jp
デフォルトゲートウェイ	133.101.32.83

---

ネットワークアプリケーションの設定は以下のようにしてください。

---

メールサーバ (POP3 サーバ)	ccpop3.kyoto-su.ac.jp
SMTP サーバ	ccmail.kyoto-su.ac.jp
NNTP ニュースサーバ	news.kyoto-su.ac.jp
WWW サーバ	www.kyoto-su.ac.jp
WWW Proxy サーバ	wwwproxy.kyoto-su.ac.jp 8000 ポート

---

## 利用上の注意と制限

307 ページの第 11.4.1 節を御一読ください。

## 11.9 トラブル対策

以下に起こりそうなトラブルについて、その原因と解決策を示しておきます。少し技術的に細かな事が書かれている場合もあるでしょう。あなたがそれを読んで判らない場合は書籍などで調べましょう。ネットニュースの [sandai.question](#) 等で質問してもよいでしょう。

また、Mule がうまく使えない場合のトラブルには 87 ページの第 4.5.2 節、特に「非 X ウインドウ環境」での起動方法とエラー対策を参考にしてください。

### 電話がかからない

接続を実行するとモデムが自動的に電話をかけます。その時に普通ならモデム<sup>28</sup> からダイヤルする音が聞こえます。普通に電話をするときに耳にする音です<sup>29</sup>。聞こえないときは説明書を見てモデムのボリュームが小さくないか確認してください。

ボリュームが大きいのに音が聞こえないか、電話をかけているとは思えない動作をしているなら二種類の原因が考えられます。

- 内線通話になっている。

外線にするために特別な操作が必要ではありませんか？0 発信ならそのように設定してください。設定個所がないようなら、電話番号の前に「0,」を挿んで「0,7053005」のようにするとうまく行くかもしれません。0 発信でない、例えば「外線ボタン」なるものがある、それを押さないといけない時はダイヤルアップ接続ができない可能性が高いです。パソコンからそのボタンを押す操作ができないからです。詳しい人を捜して相談してください。

- トーン、パルスの設定が間違っている。

トーン回線、パルス回線の指定が逆になっているかもしれません。黒電話の人は間違いなくパルスだと思われませんが、プッシュホンだからトーン回線に設定した、という人は騙されたと思って一度パルス回線に設定して試してください。

- 機器の接続ミス

モデムの電源、モデムとパソコン間のケーブル、電話コードの接続などを確認してください。

### 電話がつながらない

電話をかける音が聞こえて、ブルルルルと呼出音が鳴りガチャッと電話が受け取られた後、ピーという高い音が 2 秒程鳴り、続いてガーガーという音がしたらとりあえず電話はかけています。しかしそのまま接続に失敗するときは次のような可能性があります。

- 電話番号を間違えている。

もう一度よく確認してください。

- 電話回線に空きがない。

夜中、特に 23 時過ぎは利用者が集中するため回線に空きがない可能性があります。一度日中に試してください。

<sup>28</sup> 本体に内蔵されているパソコンなら本体から聞こえてくるように感じるかもしれません。

<sup>29</sup> 受話器を取るとツーという音が聞こえて、ダイヤルすると「びばびばびば」もしくは「じじじじじ・じじ・じじじ…」という音がするでしょう？

- パソコンの設定がおかしい。

最近のコンピュータには既に最初から設定されている項目もあります。二重に設定するとおかしくなる項目もあります。設定項目をよく確認してください。

#### telnet したがつながらない

接続は無事にできたのに telnet や WWW 接続ができない場合は、DNS の設定がうまくできていない可能性が高いです。ホスト名に cc2000.kyoto-su.ac.jp ではなく、133.101.96.17 を指定して telnet してみてください。これでつながる場合は間違いなく DNS の設定が間違っています。DNS の設定を確認してください。

インターネットにつながっている機械には世界で唯一の番号がついています。それが 133.101.96.17 のような IP アドレス と呼ばれるものなので、コンピュータはこの IP アドレス を目標にして通信しています。しかし IP アドレスはコンピュータにとっては分かりやすい番号なのですが、人間が使うときはそれでは覚えにくいので cc2000 というような ホスト名 をつけてそれを利用しています。telnet 133.101.96.17 とすればコンピュータにとって直観的に理解できて、telnet cc2000.kyoto-su.ac.jp とした場合は人間が直観的に理解できる、という訳です。

さて、telnet cc2000.kyoto-su.ac.jp としたときは逆にコンピュータはそれがどこのコンピュータなのか理解することができません。そこで、誰かが cc2000.kyoto-su.ac.jp とは 133.101.96.17 だよ、と教えてくれなくてははいけません。それをしてくれるのが DNS (Domain Name Server) なのです。

従って DNS を正しく設定していなければ cc2000.kyoto-su.ac.jp が誰なのか教えて貰えないのでつながらないという訳です。

#### アルファベットは表示されるが漢字部分の表示がおかしい

お使いの telnet ソフトの文字コード<sup>30</sup> が正しく設定されていないようです。ソフトについてくる説明書きを読んで EUC コードに設定してください。

#### Mule を使おうと思うのだが C-s , C-c <Delete>などが動作しない

お使いの telnet ソフトの設定に問題があるようです。ソフトについてくる説明書きを読んで設定してください。

#### メールソフトなどの設定がわからない

メールソフトやニュースリーダソフトは色々な種類があるため、個々に説明することができません。必要な情報は 339 ページ、第 11.8 節に書かれていますので、お使いのソフトの説明書を読んで設定してください。

#### 別に契約しているプロバイダと京産大への接続を使い分けたい

プロバイダと京産大を使い分けるには DNS の設定やメールサーバの設定をそれぞれで指定されているホスト名に設定しなければいけません。たとえばプロバイダに接続して京産大のメールサーバを使うことはできません。これを説明するのは非常に難しいので、両立したい人は自力で設定を試みてください。本

---

<sup>30</sup> 148 ページ、第 5.5.3 節参照

ガイドではサポートできません。必要な情報は 339 ページ、第 11.8 節に書かれていますので、お使いのソフトの説明書を読んで設定してください。

# 付録 A 章 リファレンス

## A.1 UNIX コマンド

ここでは UNIX のコマンドのうち、代表的なものを示します。例示は以下のような体裁をとります。

```
command : コマンドの働き (full spec of command)
command [option] string...
  option   (full spec of option) オプションの働き
  }
  string   引数の意味
  }
```

一行目にコマンド名とその概略を、二行目に実際にコマンドを実行する時のスタイルを、三行目以降に二行目で使われた記号の説明をしています。

以下にそれぞれの意味を説明します。

---

command	まさにコマンド名です。
コマンドの働き (full spec of command)	そのコマンドがどのような働きをするかを簡単に書いています。 コマンド名はその働きを示す単語の省略形である事がよくあります。その元となった単語です。
option	そのコマンドに適用できるオプションです。代表的なものだけを書いていきます。
(full spec of option)	オプション記号はその働きを示す単語の省略形である事がよくあります。その元となった単語です。
オプションの働き	そのオプションがどのような働きをするかを簡単に書いています。
string	そのコマンドに適用できる引数です。
引数の意味	与えられた引数がどのようなものとして解釈されるかを簡単に書いています。

---

オプションや引数の与え方のスタイルを示している二行目の部分では、[ ] | { } などの記号を使って表現しています。以下のルールに従って解釈してください。

- オプションや引数が [ ] で囲まれている場合がありますが、これは囲まれた部分はなくともいいという事を意味しています。  
特にオプション記号が並んでおり、それが [ ] で囲まれていた場合は囲まれた部分から任意の記号を組み合わせて書いても良い事を示しています。例えば [-abc] と書かれていた場合は -a -b -ab -abc -cb などのいずれの書き方をしても構わないのです。
- オプション記号が並んでいる時に | で仕切られている場合がありますが、これは仕切られた両側の記号のいずれか一つを選ぶ事を示しています。  
例えば [-a|-b] と書かれていた場合はオプション無しか -a か -b かいずれかで実行しなければなりません。もしも [-a|-b] ではなく {-a|-b} と書かれていた場合はオプションなしは許されず、-a か -b かいずれかを選ばなければなりません。

- 引数など斜体文字で書かれた部分については、その部分をファイル名やキーワードなど何か別の文字列で置き換えて与える事を意味しています。  
特に引数の後ろに ... と書かれていた場合は引数を空白で区切りながら複数個書いても良い事を示しています。例えば *string...* と書かれていた場合は `sample test try` と書いても構わないのです。

### A.1.1 ファイル管理に関するコマンド

**ls** : ファイルの一覧を表示する (list)

```
ls [-lagdF] [names...]
  -l      (long format)  詳細情報を表示する
  -a      (all)          隠しファイルも含めて表示する
  -g      (group)       グループ情報も表示する
  -d      (directory)   ディレクトリ以下をたどらない
  -F      (directory)   ファイルの属性が判るようにディレクトリには/を、実行ファイルには*をつける
  names   ファイル名
```

**pwd** : カレントディレクトリの表示 (print working directory)

```
pwd
```

**cd** : カレントディレクトリの移動 (change directory)

```
cd directory
  directory  移動先のディレクトリ名
```

**mkdir** : ディレクトリの作成 (make directory)

```
mkdir directory...
  directory  作成するディレクトリ名
```

**rmdir** : ディレクトリの消去 (remove directory)

```
rmdir directory...
  directory  消去するディレクトリ名
```

**cp** : ファイルのコピー (copy)

```
cp [-i|-f] source-file destination-file
cp [-i|-f] source-file... destination-directory
cp -r[i|f] source... destination-directory
  -i      (inquiry)   コピー先でファイル名が既存の場合、上書きするかを問い合わせる
  -f      (force)     コピー先でファイル名が既存であっても問い合わせをせずに上書きする
  -r      (recursive) ディレクトリ構造ごと階層的にコピーする
  source-file          コピー元ファイル
  destination-file     コピー先ファイル
  destination-directory コピー先ディレクトリ
  source               コピー元ファイルもしくはディレクトリ
```

**mv** : ファイルの移動 (move)

```
mv [-i|-f] source-file destination-file
mv [-i|-f] source... destination-directory
  -i      (inquiry)   移動先でファイル名が既存の場合、上書きするかを問い合わせる
  -f      (force)     移動先でファイル名が既存であっても問い合わせをせずに上書きする
  source-file          移動元ファイル
  destination-file     移動先ファイル
  source               コピー元ファイルもしくはディレクトリ
  destination-directory 移動先ディレクトリ
```

**rm** : ファイルの消去 (remove)

```
rm [-[r][i|f]] [name...]
  -r      (recursive) ディレクトリ構造ごと階層的に消去する
  -i      (inquiry)   消去するかどうか確認する
  -f      (force)     消去するかどうか確認しない
```

## A.1.2 ファイルに関する雑多なコマンド

**cat** : ファイルの内容を表示する (catalog)

```
cat [-n] [name...]  
-n      (number) 行番号をつける  
name    ファイル名
```

**more** : ファイルの内容を一ページずつ表示する

```
more [name...]  
name    ファイル名
```

**head** : ファイルの先頭を表示する

```
head [-number] [name...]  
-number 先頭から number 行めまで表示する  
name     ファイル名
```

**tail** : ファイルの末尾を表示する

```
tail [-number|+number] [name...]  
-number 末尾 number 行を表示する  
+number number 行めから末尾まで表示する  
name     ファイル名
```

**file** : ファイルの種類を表示する

```
file [name...]  
name    ファイル名
```

**touch** : ファイルの更新日付を新しくする

```
touch [-c] name...  
name が存在していない場合にはその名前で空ファイルを作成する。  
-c      name が存在していない場合にはファイルを作成しない。  
name    ファイル名
```

**od** : ファイルの内容をダンプする (octal dump)

```
od [-bcx] [name...]  
-b      8 進数で表示する  
-c      可視コードは文字表示する  
-x      (hex) 16 進数で表示する  
name    ファイル名
```

**split** : ファイルを行単位で分割する

```
split [-number] [name [prefix]]  
number 行ごとに name ファイルを分割し、prefix に aa から zz までアルファベット順に合成した番号を付けた複数のファイルとして出力する  
-number 分割する行単位。デフォルトでは 1000 が与えられる  
name    元ファイル名  
prefix  出力ファイルの頭に付く名前
```

**du** : ファイルの量を表示する (disk use)

```
du [-sk] [name...]  
-s      (size) name それぞれの総合計のみを表示する  
-k      (kilobyte) 表示単位をキロバイト単位にする  
name    ファイル名
```

**ln** : リンクを作成する (link)

```
ln [-fns] link [name...]  
-f      (force) リンク先 link が書き込み禁止であっても確認の問い合わせをしない  
-n      リンク名 link が既存であれば上書きしない  
-s      (symbolic link) シンボリックリンクを作成する  
link    リンク名  
name    ファイル名
```

### A.1.3 テキスト処理に関するコマンド

**wc** : ファイルの単語数などを調べる (word count)

```
wc [-clw] [name...]  
-c      (character)  文字数 (但しバイト数) を数える  
-l      (line)       行数を数える  
-w      (word)       単語数を数える  
name    ファイル名
```

**diff** : 二つのファイルの差分を表示する (difference)

```
diff [-biw] file1 file2  
-b      (ignore blank)  行末の空白を無視する  
-i      (ignore case)   大文字と小文字を区別しない  
-w      空白及びタブ文字を無視する  
file1   ファイル名  
file2   ファイル名
```

**diff3** : 三つのファイルの差分を表示する (differences of 3 files)

```
diff3 file1 file2 file3  
file1   ファイル名  
file2   ファイル名  
file3   ファイル名
```

**sort** : ファイルの内容を行単位で順に並べ変えて表示する

```
sort [-cunfr] [name...]  
sort には非常に多くのオプションがある。man sort などして確認するのが良い。  
-c      (check)        正しくソートされているか否かのチェックのみ行なう  
-u      同一行を出力しない  
-n      (number)       数値表現として解釈してソートする  
-f      大文字と小文字を無視する  
-r      (reverce)     並び順を逆にする  
name    ファイル名
```

**grep** : パターンをファイルから検索して表示する (general regular expression)

```
grep [-[AB]number] [-cinlv] pattern [name...]  
pattern には正規表現が利用できる。  
-number      マッチした行の前後 number 行を含めて表示する  
-Anumber    (after)   マッチした行の後ろ number 行を含めて表示する  
-Bnumber    (before)  マッチした行の前 number 行を含めて表示する  
-c          (count)   マッチした行数を数えるだけ  
-i          (ignore case)  大文字と小文字を無視する  
-n          (number)   マッチした行を行番号とともに表示する  
-l          少なくとも一つはマッチした行を持つファイルの名前を表示する  
-v          (invert)  マッチしなかった行を表示する  
pattern     検索するキーワード  
name       ファイル名
```

## egrep : grep の完全版 (expression grep)

egrep [-cinlv] *pattern* [*name...*]

*pattern* には全ての正規表現が利用できる。

-c	(count)	マッチした行数を数えるだけ
-i	(ignore case)	大文字と小文字を無視する
-n	(number)	マッチした行を行番号とともに表示する
-l		少なくとも一つはマッチした行を持つファイルの名前を表示する
-v	(invert)	マッチしなかった行を表示する
<i>pattern</i>		検索するキーワード
<i>name</i>		ファイル名

## fgrep : grep の高速版 (fast grep)

fgrep [-cinlv] *pattern* [*name...*]

*pattern* には正規表現は利用できない。簡略化された表現だけを提供する。

-c	(count)	マッチした行数を数えるだけ
-i	(ignore case)	大文字と小文字を無視する
-n	(number)	マッチした行を行番号とともに表示する
-l		少なくとも一つはマッチした行を持つファイルの名前を表示する
-v	(invert)	マッチしなかった行を表示する
<i>pattern</i>		検索するキーワード
<i>name</i>		ファイル名

## tr : 文字を置き換える (translate character)

tr [-ds] [*string1* [*string2*]]

*string1*, *string2* には "\012" などとして 8 進数表記も可能。[a-z] などと a から z の連続した文字を意味する省略記法も可能。

-d	(delete)	標準入力から <i>string1</i> に含まれる文字を削除して標準出力に出す
-s		標準入力の連続する <i>string1</i> を一つの <i>string2</i> に置換して標準出力に出す

## sed : ストリームエディタ (stream editor)

sed [-n] [-e *script*] [-f *scriptfile*] [*name ...*]

-n		デフォルトの出力を抑制する
-e		編集スクリプト <i>script</i> のバリエーションは多数あるのでマニュアルを参照の事
-f		編集スクリプトを <i>scriptfile</i> ファイルから読みとる
<i>name</i>		ファイル名

## spell : 単語のスペルチェックをする

spell [-bvx] [*name*]

spell コマンドは非常に高機能だがここでは基本機能のみ載せる

-b	(British)	British 綴りをチェックする
-v	(verbose)	その綴りでスペリングリストに入っていないすべての単語を表示し、その単語から可能性のある派生語が示す
-x		可能性のある語幹を等号 (=) と共に表示する
<i>name</i>		ファイル名

## A.1.4 プリンタに関するコマンド

プリンター一覧

**lpr** : プリンタに出力する (line printer)

`lpr [-Pprinter] [name...]`

cc 環境でプリンタに出力できる内容は単純なテキストファイルか、PostScript ファイルに限られる。

`-Pprinter` `printer` で示されるプリンタに出力する  
`name` ファイル名

**lpq** : プリント待ちキューの内容を表示する (line printer queue)

`lpq [-Pprinter]`

`-Pprinter` `printer` で示されるプリンタの待行列を表示する

**lprm** : プリント待ちエントリを消去する (line printer remove entry)

`lpq [-Pprinter] [-] [job...]`

`job` は `lpq` コマンドであらかじめ確認する。

`-Pprinter` `printer` で示されるプリンタのエントリを処理対象にする  
`-` 自分が最後に出力したエントリを消去する  
`job` `job` で示されるエントリを消去する

## A.1.5 アクセス権、アクセス制御に関するコマンド

**whoami** : 現在の利用者名を表示する (who am i)

`whoami`

**id** : 現在の利用者名、現在のグループ名などを表示する

`id`

`id` コマンドは各 OS によって相違が見られるので、`man id` などして確認するのが良い。

**groups** : 所属するグループ名の一覧を表示する

`groups [username...]`

`username` 表示させたい利用者名。デフォルトでは現在の利用者の利用者名が与えられる

**newgrp** : 新しいグループへのログイン (new group)

`newgrp [group]`

`group` グループ名。デフォルトでは現在の利用者のデフォルトグループが与えられる

**chmod** : ファイルのアクセス権を変更する (change mode)

`chmod [-R] mode name...`  
-R (recursive) *name* がディレクトリであった場合は階層的に処理を行なう  
*mode* 下部参照  
*name* ファイル名

相対指定における *mode* は {u|g|o|a}{+|-|=}{r|w|x} となる。

対象	オペレータ	設定内容
u 所有者	+ 追加	r 読みだし
g グループ	- 取消	w 書き込み
o その他の人	= 強制	x 実行
a 全ての人		

絶対指定における *mode* は以下の数値の和となる。

値	対象	設定内容	値	対象	設定内容	値	対象	設定内容
0400	所有者	読みだし	0040	グループ	読みだし	0004	その他の人	読みだし
0200	所有者	書き込み	0020	グループ	書き込み	0002	その他の人	書き込み
0100	所有者	実行	0010	グループ	実行	0001	その他の人	実行

**chgrp** : ファイルのグループを変える (change group)

`chgrp [-R] group name...`  
-R (recursive) *name* がディレクトリであった場合は階層的に処理を行なう  
*group* 変更したいグループ名  
*name* ファイル名

## A.1.6 マニュアルに関するコマンド

**man** : コマンドなどのマニュアルを表示する (manual)

`man [-s section] title`  
`man -k keyword...`  
-s *section* *title* のセクションを *section* に限定する  
-k *keyword* *keyword* にマッチするタイトルの一覧を表示する  
*keyword* はコマンド名などの一部でも良い  
*title* コマンド名など

**whatis** : コマンドなどの要約を表示する (what is)

`whatis title...`  
*title* コマンド名などのキーワード

**which** : コマンドファイルの位置を表示する

`which command...`  
*command* コマンド名

**whereis** : コマンドファイル、マニュアルファイルの位置を表示する

`whereis command...`  
*command* コマンド名

## A.1.7 雑多なコマンド

**date** : 現在の日付を表示する

`date`

**sleep** : 実行を保留する

`sleep [time]`  
*time* 待ち時間 (秒)

**cal** : カレンダーを表示する (calender)

`cal` *[[month] year]*

オプションを全て省略すると今月のカレンダーを表示する。*year* だけを与えるとその年のカレンダーを表示する。*month* も与えるとその年のその月のカレンダーを表示する。

*month* 月の指定を 1 から 12 までで与える

*year* 年の指定を西暦で与える

**echo** : 引数を表示する

`echo` *[-n] [argument...]*

*-n* (no return) 出力に改行文字を加えない

*argument* 引数

**banner** : 引数を花文字で表示する

`banner` *string...*

*string* 10 文字までの花文字にしたい文字列

**clear** : 画面を消去する

`clear`

**bc** : 電卓

`bc` *[name...]*

*name* ファイル名。但しファイルの最後には `quit` を `bc` へのサブコマンドとして書く

**look** : 英単語を辞書検索する

`look` *[-df] [-tc] string*

*-d* (dictionary order) 文字、数字、タブ及びスペースが比較される

*-f* (fold case) 大文字と小文字を区別しない

*-tc* (termination character) *c* とその後に続く文字を無視する

*string* 検索単語

**tee** : 標準入力を標準出力とファイルに書く

`tee` *[-ai] [name...]*

*-a* (append) ファイルへの出力を追加書きにする

*-i* (ignore interrupts) 割り込みを無視する

*name* ファイル名

**script** : 端末の操作をファイルに記録する

`script` *[-a] [name]*

`script` コマンドは新しくシェルを起動する。このシェルを終了することによって `script` コマンドも終了する。シェルを終了するには `exit` コマンドを利用すれば良い。

*-a* (append) *name* ファイルに対する出力を追加書きで行なう。デフォルトは上書き

*name* ファイル名。省略するとデフォルトとして `typescript` が与えられる

**df** : 利用可能なディスクブロックを表示する (disk free)

`df` *[directory...]*

`df` コマンドは各 OS によって相違が見られるので、`man df` などして確認するのが良い。

*directory* *directory* が存在するファイルシステムだけに限定して表示する

**yppasswd** : パスワードを変更する (YP password)

`yppasswd`

**ypchfn** : フルネームを変更する (YP change full name)

`ypchfn`

`ypchfn` コマンドは Sun OS 独自のコマンドで、`cc` 環境では `cc2000` でしか動かない。これによってメールの From: 行や `finger` などの表示に現れるローマ字の名前を変更できる。

**ypchsh** : login シェルを変更する (YP change shell)

`ypchsh`

`ypchsh` コマンドは Sun OS 独自のコマンドで、`cc` 環境では `cc2000` でしか動かない。これによって `login` した後に実行されるシェルを変更できる。この表現が理解できない場合はこのコマンドは実行しない方がよい。変更できるシェルには限りがあり、`cat /etc/shells <return>` などして確認できる。

## A.1.8 ファイル圧縮などに関するコマンド

**tar** : テープ用集積ファイル (tarfile) を扱う (tape archiver)

`tar [-] [c|r|t|u|x] [v] [-b block] [-f device] [name...]`

テープにファイルを書き込む時などに使うが、ブロックサイズなどはテープドライブに依存するので各デバイスなどのマニュアルで確認する事。

-c	(creat)	tarfile を <i>name</i> から作成する
-r	(replace)	tarfile に <i>name</i> を追加する
-t	(title)	tarfile の内容一覧を表示する
-u	(update)	tarfile を更新する。最後に tarfile へ <i>name</i> ファイルを書き込んでから変更があれば tarfile に追加する
-x	(extract)	tarfile からファイル <i>name</i> を抽出もしくはレストアする
-v	(verbose)	現在の状態を表示しながら実行する
-b	(block size)	ファイルのブロック化係数を <i>block</i> にする
-f	(device file)	tarfile を格納するデバイスを <i>device</i> にする
<i>block</i>		ブロックサイズ。デフォルトは 20
<i>device</i>		デバイスファイル名。-で標準入出力、通常のファイル名の指定も可能
<i>name</i>		ファイル名もしくはディレクトリ名

**uuencode** : バイナリファイルを可視コード文字列に変換する

`uuencode [name] label`

結果は標準出力に出る。*name* を省略すると標準入力から読んだデータを変換する。

<i>name</i>	ファイル名
<i>label</i>	変換したファイルを uuencode コマンドで戻す時のファイル名

**uudecode** : uuencode によって変換されたファイルをバイナリファイルに逆変換する

`uudecode [name]`

uuencode 時の *label* によって指定されたファイル名で結果が作成される。*name* を省略すると標準入力から読んだデータを変換する。

<i>name</i>	ファイル名
-------------	-------

**compress** : ファイルを圧縮する

`compress [-cv] [name...]`

圧縮されたファイルは *name.Z* という名前で作成され、元の *name* ファイルは消去される。

-c		圧縮結果を <i>name.Z</i> ファイルに作成せずに標準出力に出力する <i>name</i> ファイルは消去されない
-v	(verbose)	圧縮率を表示する
<i>name</i>		ファイル名

**uncompress** : compress コマンドで圧縮されたファイルを復元する

`uncompress [-cv] [name...]`

*name* は最後が.Z でなければならない。復元されたファイルは *name* から.Z が外された名前で作成され、元の *name* ファイルは消去される。

-c		復元結果を <i>name</i> から.Z を取り除いたファイルに作成せずに標準出力に出力する <i>name</i> ファイルは消去されない
-v	(verbose)	圧縮率を表示する
<i>name</i>		ファイル名

**gzip** : ファイルを圧縮する (GNU zip)

`gzip [-cdhlv] [name...]`

圧縮されたファイルは *name.z* という名前で作成され、元の *name* ファイルは消去される。

-c		圧縮結果を <i>name.z</i> ファイルに作成せずに標準出力に出力する <i>name</i> ファイルは消去されない
-d	(decompress)	圧縮ファイルを復元する
-h	(help)	オプション一覧を表示する
-l	(list)	圧縮ファイルの内容一覧を表示する
-v	(verbose)	圧縮率を表示する
<i>name</i>		ファイル名

**gunzip** : gzip および compress コマンドで圧縮されたファイルを復元する (GNU unzip)

```
gunzip [-cv] [name...]  
name は最後が.Z もしくは.z でなければならない。復元されたファイルは name から.Z もしくは.z  
が外された名前で作成され、元の name ファイルは消去される。  
-c                復元結果をファイルに作成せずに標準出力に出力する  
                   name ファイルは消去されない  
-d (decompress)  圧縮ファイルを復元する  
-h (help)         オプション一覧を表示する  
-l (list)         圧縮ファイルの内容一覧を表示する  
-v (verbose)     圧縮率を表示する  
name             ファイル名
```

**zcat** : compress コマンドで圧縮されたファイルを表示する (cat Z file)

```
zcat [name...]  
name は最後が.Z でなければならない。gzip 圧縮ファイルを扱える場合もある。gunzip 同様のオプ  
ションが使える場合もある。  
name          ファイル名
```

### A.1.9 プロセスに関するコマンド

**ps** : 現在処理中のプロセス一覧を表示する (process)

```
ps [[-]aux]  
ps コマンドはオプションもその表示も各マシン、その OS 種類によって大きく異なる。man ps など  
して確認するのが良い。  
a (all)   全ての利用者のプロセスを表示する  
u (user)  利用者情報指向に整形して表示する  
x        端末制御を持たないプロセスも含めて表示する
```

**kill** : プロセスを終了させる

```
kill [-l] [-signal] process-id...  
process-id についてはあらかじめ ps コマンドで確認しておく。signal の種類によっては終了ではな  
くプロセスの再起動などが行なわれる場合がある。  
-signal    process-id によって示されるプロセスに対して送られるシグナルの種類。  
1          signal に利用可能な記号の一覧を表示する  
           デフォルトでは-15 (-TERM) が与えられ、大抵これで終了させられる。強制終了の為  
           には-9 (-KILL) を与える  
process-id 終了させたいプロセスの番号
```

### A.1.10 現在使っているコンピュータに関するコマンド

**tty** : 端末回線名を表示する (tele type terminal)

```
tty
```

**hostname** : ホスト名を表示する (host name)

```
hostname
```

**uname** : OS に関する情報を表示する

```
uname [-apsv]  
-a (all)   全ての情報を表示する  
-p (processor) プロセッサ型を表示する  
-s (operating system) OS 名を表示する。これはデフォルトで与えられる  
-v (version) OS のバージョンを表示する
```

**uptime** : 起動されてからの時間と CPU 負荷率を表示する

```
uptime
```

## A.1.11 利用者に関するコマンド

**who** : 現在利用している利用者の一覧を表示する

**who** [-Hq]  
-H (header) 見出しを出力する  
-q 簡略化された形式で出力する

**w** : 現在利用している利用者と作業内容の一覧を表示する

**w** [-hls] [*username*]  
見出しに `uptime` コマンドで表示されるのと同じ CPU 負荷率などが表示される  
-h 見出しを表示しない  
-l (long format) 長い出力形式。これはデフォルトで与えられる  
-s (short format) 短い出力形式  
*username* 利用者名 *username* に関する情報だけに限定する

**finger** : 利用者情報を表示する

**finger** [-lms] [*keyword*...]

**finger** [-l] [*username*]@*hostname*...

`finger` コマンドによって表示される最後にメールを読んだ日付については `cc` 環境では正しく表示されない場合がありますので無視してください。

-l (long format) 長い出力形式  
-m *keyword* を利用者名に限定して検索する  
-s (short format) 短い出力形式  
*keyword* 利用者名、氏名などの断片  
*username* 利用者名。省略した場合は *hostname* コンピュータを利用している利用者一覧が表示される  
*hostname* リモートコンピュータのホスト名

**whois** : 利用者情報を表示する (who is)

`whois [-h hostname] keyword`

`whois` サービスの内容や使い方はそれが行なわれているサーバに大きく依存する。

`-h hostname` *hostname* コンピュータを whois サーバとする  
`keyword` 検索するキーワード

## A.1.12 ネットワークサービスに関するコマンド

**telnet** : TELNET プロトコルによるリモートログインを行なう

`telnet [hostname]`

*hostname* リモートログインするホスト名

**rlogin** : リモートログインを行なう (remote login)

`rlogin [-8] [-1 username] hostname`

`-8` (8 bit) 通信に 8 ビットを利用する。デフォルトは 7 ビットの可能性が高い。

`-1 username` リモートログイン用の利用者名として *username* を使う

デフォルトでは現在の利用者名が与えられる

*hostname* リモートログインするホスト名

**rsh** : リモートマシンにコマンドを実行させる (remote shell)

`rsh [-1 username] hostname command`

`-1 username` リモートログイン用の利用者名として *username* を使う

デフォルトでは現在の利用者名が与えられる

*hostname* *command* を実行させるホスト名

*command* 実行させたいコマンド行

**rcp** : リモートマシンのファイルをコピーする (remote copy)

`rcp [[username@]hostname:]source-file [[username@]hostname:]destination`

`rcp -r [[username@]hostname:]source... [[username@]hostname:]destination-directory`

コピー元、先のファイル名の記述の先頭に *hostname:* を付加する事によって「:」以降に記述されているファイルはそのホストに存在する事を意味する。*hostname:* を省略すればデフォルトとして現在のホスト名が与えられる。更に *username:* を与える事によって「@」以降に記述されているホストに対するアクセスは利用者名 *username* で行なわれる事を意味する。*username:* を省略すればデフォルトとして現在の利用者名が与えられる。

`-r` (recursive) ディレクトリ構造ごと階層的にコピーする

*username* @以降に記述されるホストでの利用者名

*hostname* :以降に記述されるファイルが存在するホスト名

*source-file* コピー元ファイル

*destination* コピー先ファイルもしくはディレクトリ

*destination-directory* コピー先ディレクトリ

*source* コピー元ファイルもしくはディレクトリ

### A.1.13 シェル (tcsh) のサブコマンド

cd, which, kill などシェルの子コマンドだが、これらは普通のコマンドとして紹介している。ここでの記述は tcsh 特有のものを含んでいる事に注意。

#### シェル変数、環境変数に関するサブコマンド

**set** : シェル変数を定義する

set [variable[=string]]

引数なしで現在定義されているシェル変数を表示する。variable を与えながら string を省略すると variable で示されるシェル変数を空文字列とする。

variable シェル変数名  
string 文字列

**unset** : シェル変数の定義を解除する

unset variable

variable シェル変数名

**setenv** : 環境変数を定義する (set environment variable)

setenv [variable [string]]

引数なしで現在定義されている環境変数を表示する。variable を与えながら string を省略すると variable で示されるシェル変数を空文字列とする。

variable 環境変数名  
string 文字列

**unsetenv** : 環境変数の定義を解除する (unset environment variable)

unsetenv variable

variable 環境変数名

#### ジョブ制御に関するサブコマンド

**jobs** : 現在実行中のコマンドの一覧を表示する

jobs [-l]

l (long) プロセス番号も表示する

**fg** : 停止中のジョブを再開する (fore ground)

fg [%job]

job jobs コマンドで表示されたジョブ番号

**bg** : 停止中のジョブをバックグラウンドで再開する (back ground)

bg [%job]

job jobs コマンドで表示されたジョブ番号

**stop** : バックグラウンドで実行中のジョブを一時停止する

stop [%job]

job jobs コマンドで表示されたジョブ番号

**notify** : バックグラウンドで実行中のジョブの状態変化を知らせる

notify [%job]

job jobs コマンドで表示されたジョブ番号

**wait** : 全てのバックグラウンドジョブの実況終了を待つ

wait

## %job 番号の指定方法

100	プロセス番号 100 番
%1	ジョブ番号 1 番
%	直前に操作したジョブ
%-	一つ前のジョブ
%cc	実行コマンドが cc で始まるジョブ
/?sort	実行コマンドに sort を含むジョブ

プロセス番号は ps コマンドもしくは jobs -l コマンドで確認できる  
ジョブ番号は jobs コマンドで確認できる

## 雑多なサブコマンド

**alias** : コマンドの別名を定義する

`alias [name [string]]`

引数なしで現在定義されているエイリアス一覧を表示する。*name* を与えながら *string* を省略すると *name* で定義されているエイリアスを表示する。

*name*       エイリアス名  
*string*      定義する文字列

**unalias** : エイリアスの定義を解除する

`unalias name`

*name*       エイリアス名

**rehash** : コマンド参照の為に内部ハッシュテーブルを更新する (re-assign hash table)

`rehash`

**unhash** : コマンド参照の為に内部ハッシュテーブルを使わなくする (unuse hash table)

`unhash`

**login** : login シェルを終了し、新たに login する

`login`

**logout** : login シェルを終了する

`logout`

**exit** : シェルを終了する

`exit (expr)`

*expr* で与えられた数値は \$status シェル変数に与えられる  
*expr*       数値もしくは数値になる式。() は省略できる

**exec** : コマンドを実行する (execute)

`exec name`

*name* を実行する。実行が終っても制御は返ってこない。  
*name*       コマンド名もしくは実行可能なファイル名

**source** : 実行するコマンドの指定をファイルから読む

`source name`

ファイル名 *name* に書かれたコマンドを現在のシェルで実行する。  
*name*       ファイル名

**history** : 実行したコマンドの履歴を見る

`history [-hr] number`

ファイル名 *name* に書かれたコマンドを現在のシェルで実行する。

*h*           イベント番号を表示に付けない  
*r*           (reverse) 履歴から最新の *number* 個のイベントを逆順に表示する  
*number*     表示するイベントの数

## A.1.14 索引

コマンド	頁 ( 章節 )	コマンド	頁 ( 章節 )	コマンド	頁 ( 章節 )
alias	357 ( A.1.13 )	jobs	356 ( A.1.13 )	split	346 ( A.1.2 )
banner	351 ( A.1.7 )	kill	353 ( A.1.9 )	stop	356 ( A.1.13 )
bc	351 ( A.1.7 )	ln	346 ( A.1.2 )	tail	346 ( A.1.2 )
bg	356 ( A.1.13 )	login	357 ( A.1.13 )	tar	352 ( A.1.8 )
cal	351 ( A.1.7 )	logout	357 ( A.1.13 )	tee	351 ( A.1.7 )
cat	346 ( A.1.2 )	look	351 ( A.1.7 )	telnet	355 ( A.1.12 )
cd	345 ( A.1.1 )	lpq	349 ( A.1.4 )	touch	346 ( A.1.2 )
chgrp	350 ( A.1.5 )	lpr	349 ( A.1.4 )	tr	348 ( A.1.3 )
chmod	350 ( A.1.5 )	lprm	349 ( A.1.4 )	tty	353 ( A.1.10 )
clear	351 ( A.1.7 )	ls	345 ( A.1.1 )	unalias	357 ( A.1.13 )
compress	352 ( A.1.8 )	man	350 ( A.1.6 )	uname	353 ( A.1.10 )
cp	345 ( A.1.1 )	mkdir	345 ( A.1.1 )	uncompress	352 ( A.1.8 )
date	350 ( A.1.7 )	more	346 ( A.1.2 )	unhash	357 ( A.1.13 )
df	351 ( A.1.7 )	mv	345 ( A.1.1 )	unset	356 ( A.1.13 )
diff	347 ( A.1.3 )	newgrp	349 ( A.1.5 )	unsetenv	356 ( A.1.13 )
diff3	347 ( A.1.3 )	notify	356 ( A.1.13 )	uptime	353 ( A.1.10 )
du	346 ( A.1.2 )	od	346 ( A.1.2 )	uudecode	352 ( A.1.8 )
echo	351 ( A.1.7 )	ps	353 ( A.1.9 )	uuencode	352 ( A.1.8 )
egrep	348 ( A.1.3 )	pwd	345 ( A.1.1 )	w	354 ( A.1.11 )
exec	357 ( A.1.13 )	rcp	355 ( A.1.12 )	wait	356 ( A.1.13 )
exit	357 ( A.1.13 )	rehash	357 ( A.1.13 )	wc	347 ( A.1.3 )
fg	356 ( A.1.13 )	rlogin	355 ( A.1.12 )	whatis	350 ( A.1.6 )
fgrep	348 ( A.1.3 )	rm	345 ( A.1.1 )	whereis	350 ( A.1.6 )
file	346 ( A.1.2 )	rmdir	345 ( A.1.1 )	which	350 ( A.1.6 )
finger	354 ( A.1.11 )	rsh	355 ( A.1.12 )	who	354 ( A.1.11 )
grep	347 ( A.1.3 )	script	351 ( A.1.7 )	whoami	349 ( A.1.5 )
groups	349 ( A.1.5 )	sed	348 ( A.1.3 )	whois	355 ( A.1.11 )
gunzip	353 ( A.1.8 )	set	356 ( A.1.13 )	ypchfn	351 ( A.1.7 )
gzip	352 ( A.1.8 )	setenv	356 ( A.1.13 )	ypchsh	351 ( A.1.7 )
head	346 ( A.1.2 )	sleep	350 ( A.1.7 )	yppasswd	351 ( A.1.7 )
history	357 ( A.1.13 )	sort	347 ( A.1.3 )	zcat	353 ( A.1.8 )
hostname	353 ( A.1.10 )	source	357 ( A.1.13 )		
id	349 ( A.1.5 )	spell	348 ( A.1.3 )		

## A.2 UNIX でよく使われる記号など

ここでの記述は tcsh 特有のものを含んでいる事に注意。

### A.2.1 シェル変数の一覧

argv	シェルスクリプトに渡された引数の列
status	直前のコマンド終了時の戻り値を示す
cwd	Current Working Directory. カレントディレクトリを示す
home	ホームディレクトリを示す
path	コマンドパスを示す。set サブコマンドで設定可
user	現在のシェルプロセスのユーザ ID を示す
uid	現在のシェルプロセスのユーザ id を示す
gid	現在のシェルプロセスのグループ id を示す
term	現在利用しているターミナル種別を示す。set サブコマンドで設定可
tty	現在利用しているターミナルの回線番号を示す
prompt	シェルプロンプトの形式を示す。set サブコマンドで設定可
autologout	セットした場合、この秒数の間入力がなければ tcsh は自動的に終了する
ignoreeof	セットした場合、端末から C-d を読んでもシェルは終了しない。終了には exit を使う
noclobber	セットした場合、シェルは既存のファイルに出力をリダイレクトする事を許さない。この設定を無視してコマンドを実行するには!を使う
noglob	セットした場合、*,?などのワイルドカードによるファイル名の補間を行わない
nonomatch	セットした場合、ファイル名の補間に失敗してもエラーとせずコマンドを起動する
verbose	セットした場合、シェルはエイリアス、コマンド、ファイル名、変数などの置換えをした後のコマンドを表示しながら実行する
history	コマンド履歴の最大数を示す。set サブコマンドで設定可
histchars	履歴置き換え文字を示す。未設定の場合!を使う
savehist	ファイルに残すコマンド履歴の最大数を示す。set サブコマンドで設定可
shell	現在のシェルを示す
tcsh	tcsh のバージョンを示す
version	現在の tcsh のバージョンを示す

### A.2.2 環境変数の一覧

PATH	コマンドパスを示す。path シェル変数と連動している
HOME	ホームディレクトリを示す
PWD	カレントディレクトリを示す
SHELL	現在のシェルを示す
HOST	ホスト名を示す
HOSTTYPE	ホストコンピュータの種別を示す
LOGNAME	現在のシェルプロセスのユーザ ID を示す
USER	現在のシェルプロセスのユーザ ID を示す
LANG	言語環境を示す
TERM	現在利用しているターミナル種別を示す。term シェル変数と連動している
MANPATH	man コマンドが検索するマニュアルファイルの置き場所を示す
DISPLAY	X ウィンドウアプリケーションの表示画面先を示す
EDITOR	標準のエディタを示す
PAGER	標準のページャを示す
PRINTER	標準のプリンタ名を示す
TEXTFONTPATH	T <sub>E</sub> X のフォントディレクトリを示す
TEXTFMPATH	T <sub>E</sub> X のフォントディレクトリを示す
ARCH	ホストコンピュータのアーキテクチャを示す。cc 環境特有
ENVIRON	ホストコンピュータの環境種別を示す。cc 環境特有
XENVIRON	ホストコンピュータの X ウィンドウの環境種別を示す。cc 環境特有
COMMON	共通設定ディレクトリを示す。cc 環境特有

### A.2.3 リダイレクション記号など

<	標準入力をファイルから読む
<< string	文字列 string が入力行の先頭に現れるまで標準入力を読む
>	標準出力をファイルに書く
>>	標準出力をファイルに追加書きする
>&	エラー出力をファイルに書く
>>&	エラー出力をファイルに追加書きする
>!	標準出力をファイルに書く (noclobber シェル変数による保護を無視)
>>!	標準出力をファイルに追加書きする (noclobber シェル変数による保護を無視)
>&!	エラー出力をファイルに書く (noclobber シェル変数による保護を無視)
>>&!	エラー出力をファイルに追加書きする (noclobber シェル変数による保護を無視)
	標準出力をパイプに書く
&	エラー出力もパイプに書く

#### リダイレクション記号などを使ったコマンド実行の例

command	通常のコマンド実行
command &	バックグラウンドでのコマンド実行
command1 ; command2	command1 の実行が済めば command2 を実行する
(command1 ; command2)	command1 ; command2 に同じ。但し単一コマンドのようにシェルは扱う
command1   command2	通常のパイプ付き実行
command1  & command2	エラー出力を含めたパイプ付き実行
command1 && command2	command1 の実行が成功すれば command2 を実行する
command1    command2	command1 の実行が失敗すれば command2 を実行する
(command > outfile) >& errorfile	標準出力とエラー出力を分ける

### A.2.4 ファイル指定のワイルドカードなど

*	任意のゼロ個以上の文字
?	任意の一文字
[characters]	[] に囲まれた文字列 characters に含まれる任意の一文字
[char1-char2]	文字 char1 から char2 までの範囲に含まれる任意の一文字
{string1,string2,...}	文字列 string1 もしくは string2 などのいずれか
.	カレントディレクトリ
..	カレントディレクトリの一つ上の階層のディレクトリ
~	自分のホームディレクトリ
~username	ユーザ IDusername のホームディレクトリ

### A.2.5 コマンド履歴を扱う為の表記法

!!	直前のコマンド行
!n	n 番目のコマンド行
!-n	n 番前のコマンド行
!string	string から始まる最近のコマンド行
!?string	string を含む最近のコマンド行
!\$	直前のコマンド行の最後の単語
!*	直前のコマンド行の 1 番めから最後の単語 (つまりコマンド以外の全ての引数)
!n:\$	n 番目のコマンド行の最後の単語
!n:^	n 番目のコマンド行の最初の単語
!n:m	n 番目のコマンド行の m 番目の単語
!n:m-1	n 番目のコマンド行の m 番めから 1 番めの単語
!n:*	n 番目のコマンド行の 1 番めから最後の単語 (つまりコマンド以外の全ての引数)
~str1^str2^	直前のコマンドの str1 を str2 に置き換える

## A.2.6 正規表現

a	a (通常の文字) にマッチする
.	任意の1文字にマッチする
^	行頭にマッチする
\$	行末にマッチする
^.....\$	5文字の行にマッチする
	前後の正規表現のいずれかにマッチする
ab cd	ab もしくは cd のどちらかにマッチする
ab cd ef	ab、cd もしくは ef のいずれかにマッチする
(ab cd)(12 34)	ab12 ab34 cd12 cd34 のいずれかにマッチする
[abc]	abc のどれか1文字にマッチする
[^abc]	abc のどれか1文字以外にマッチする
[a-z]	a から z までの範囲のどれか1文字にマッチする
[a-hxyz0-9]	abcdefghxyz0123456789 のどれか1文字にマッチする
?	?直前の正規表現のゼロ個ないしは一個にマッチする
ab?c	ac abc にマッチする。abbc などにはマッチしない
*	*直前の正規表現のゼロ個以上の繰り返しにマッチする
ab*c	ac abc abbbc などにマッチする。ab1c などにはマッチしない
a.*c	ac abc abbbc a123c などにマッチする
+	+直前の正規表現の一個以上の繰り返しにマッチする
ab+c	abc abbc abbbc などにマッチする。ac ab1c などにはマッチしない
a.+c	abc abbc abbbc a1c などにマッチする。ac にはマッチしない
\{num\}	直前の正規表現の num 個の繰り返しにマッチする
\{num,\}	直前の正規表現の num 個以上の繰り返しにマッチする
\{num1,num2\}	直前の正規表現の num1 個から num2 個までの繰り返しにマッチする

\ ^ \$ . [ ] ( ) | \* + ? は普通の文字ではなく、意味を持ったメタキャラクタである。メタキャラクタに使われている記号をそのまま表現したい時は\\$などのように \ 記号に続けて書く。 \ 記号を表現したい時は\\である。

## A.3 Mule コマンド

### 表記方法

C-h	コントロールキーを押しながら h を押す
M-x ABC	エスケープキーを押した後に x を押して普通に ABC と打つ
M-C-x	エスケープキーを押した後にコントロールキーを押しながら x を押す
<Space>	スペースキーを押す
<return>	リターンキーを押す
<Delete>	デリートキーを押す

### A.3.1 絶対覚えておいた方がいいもの

mule <return>	Mule を起動する
C-x C-c	Mule を終了する
C-h T Japanese <return>	Mule(日本語) のチュートリアルを表示する
C-x C-f <i>filename</i> <return>	ファイルを読み込む
C-x C-w <i>filename</i> <return>	ファイル名を変更して保存する
C-g	指示途中のコマンド操作を取り消す
C-l	カーソルのある行を中央へ移動する
C-x u または C-_	直前の編集操作を取り消す (Undo)
M-x goto-line <return>	指定した行にジャンプする

### カット&ペースト

C-<Space>	カーソルの位置にマークをセットする
C-x C-x	カーソルの位置とマークの位置を入れ替える
C-w	マークの位置からカーソルの前までを記憶して消去する (カット)
M-w	マークの位置からカーソルの前までを記憶する (コピー)
C-y	記憶した文字列をカーソルの位置に挿入する (ペースト)
C-k	カーソルの位置から行末までを記憶して消去する

### 検索

C-s <i>String</i>	カーソル位置より下方向に向かって検索する
C-r <i>String</i>	カーソル位置より上方向に向かって検索する
C-s	下方向に検索を続ける
C-r	上方向に検索を続ける
C-g	検索を終了しカーソルを検索開始前位置に戻す

### Wnn

C-\	Wnn を起動 / 終了する
<Space>	変換する、次候補を表示する (C-n でも可)
C-p	前候補を表示する
<return>	変換文字を確定する
C-o	文節を伸ばす
C-i	文節を縮める
C-f	右の文節へ移動する
C-b	左の文節へ移動する

## カーソル操作

		先頭	M-<			
		前ページ	M-v			
		1行上	C-p			
行頭	1語前	1字前		1字後	1語後	行末
C-a	M-b	C-b		C-f	M-f	C-e
		1行下	C-n			
		次ページ	C-v			
		末尾	M->			

### A.3.2 必要に応じて覚えるもの

#### 起動時

mule <i>filename</i> <return>	Mule を起動して <i>filename</i> をバッファに読み込む
mule -q	.emacs を無視して Mule を起動する
mule -u <i>usr_name</i>	<i>usr_name</i> の人の .emacs の設定で Mule を起動する

#### ファイル操作

C-x i <i>filename</i> <return>	別ファイルをカーソルの位置に差し込む
C-x C-s	カレントバッファを保存する
C-x s	すべてのバッファを保存する
C-x k	カレントバッファを保存せずにクローズする
C-x C-v	カレントバッファにファイルを読み込む (カレントバッファの内容はクローズする)
C-x b	バッファを切替える
C-x C-b	バッファリストを表示する
C-x C-q	書き込みモードを変更する
M-x recover-file <return>	autosave された内容を読み込む

#### コードの変更

C-x C-k d	画面入出力コード変更
C-x C-k i	キーボードからの入力コード変更
C-x C-k f	ファイルの入出力コード変更

## その他

### 全角、半角文字

M-x zenkaku-region リージョン範囲を全角にする  
M-x hankaku-region リージョン範囲を半角にする

### 確定後の再変換

M-x henkan-region リージョン範囲を変換する。  
M-x gyaku-henkan-region リージョン範囲の漢字をひらがなに戻す。  
M-x roma-kana-region リージョン範囲のローマ字をひらがなにする

### 文字の入れ換え

C-t カーソルの位置の文字とその左の文字を入れ換える

### 繰り返し

C-u *n Command* *Command* を *n* 回繰り返す  
または M-*n Command* 例: C-u 5 C-\_ アンドウ5回

## カーソル操作

M-a 文の先頭へ移動する  
M-e 文の末尾へ移動する

### カーソルの位置

M-x what-line <return> 今カーソルが何行目にあるか表示する  
C-x l 全部の行数と現在のカーソル位置を表示する

### 改ページ

C-q C-l 改ページ文字<sup>^</sup>Lを入力する

## 消去、カット&ペースト

M-x kill-rectangle <return> マークセット位置からカーソル位置までのブロックを消去する  
M-x clear-rectangle <return> マークセット位置からカーソル位置までのブロックを空白に置換する  
M-x yank-rectangle <return> 消去したブロックをカーソル位置に挿入する

## 置換

M-% *search-string* <return> *change-string* <return>  
検索文字列を置換文字列に確認しながら置換する  
<Space>または y で置換を行なう  
<Delete>または n で置換を行なわない  
! で残り全部を確認せずに置換を行なう  
^ で一つ前にもどる  
M-x replace-string <return> *search-string* <return> *change-string* <return>  
検索文字列を置換文字列にすべて置換する

## 画面分割

C-x 2	上下に二分割する
C-x 3	左右に二分割する (Emacs では C-x 5)
C-x o	カーソルを別ウインドウに移動する
C-x 1	カーソルのあるウインドウ以外のウインドウを隠す
C-x 0	カーソルのあるウインドウを隠す

## ウインドウのリサイズ

C-x ^	カーソルのあるウインドウを縦方向に拡大する
C-x }	カーソルのあるウインドウを横方向に拡大する

## バッファリスト

C-x C-b	バッファ一覧を表示する
?	バッファリスト簡易ヘルプを表示する
f	カーソルの行のバッファをウインドウに表示する
l	カーソルの行のバッファだけをウインドウに表示する
q	バッファ一覧を終了する

## ヘルプ

C-h	ヘルプを呼び出す
C-h C-h C-h	ヘルプオプションとその説明を表示する
C-h k <i>Command</i>	<i>Command</i> の引数説明を表示する
C-h a <i>String</i>	<i>String</i> を含むコマンドの一覧を表示する
C-h b	現在のキー割当を表示する

## オンラインマニュアル

C-h i	オンラインマニュアルを起動する
q	オンラインマニュアルを終了する
m	メニューを選択する
u	前のメニューに戻る
<Space>	続きを読む
<Delete>	前に戻る
n	次の項目に進む
p	前の項目に戻る
d	オンラインマニュアルの最初のメニューに戻る

## マクロ

C-x (	キーボード操作記憶開始
C-x )	キーボード操作記憶終了
C-x e	記憶したキーボード操作実行
M-x name-list-kbd-macro <return> <i>macroname</i> <return>	マクロに名前を付ける
M-x <i>macroname</i>	マクロ実行
M-x insert-kbd-macro <return> <i>macroname</i> <return>	マクロ定義ファイル書き出し
M-x local-set-key C-c C-d ' <i>macroname</i>	マクロを C-c C-d に定義

## Wnn

### 変換

M-h	(変換途中で) ひらがなにする
M-k	(変換途中で) カタカナにする
q	アルファベット入力モードにする
C-q	アルファベット入力モードをやめる
C-k または C-c	変換をキャンセルする (ひらがなに戻る) (C-f C-b で前後して訂正可能)

### 変換候補の一覧

M-s	変換候補一覧をエコー行に表示する
C-n	次の一覧部分を表示する
C-p	前の一覧部分を表示する

### 記号の入力

C-^	記号一覧のメニューをエコー行に表示する
または	0. JIS 入力
M-x special-symbol-input	1. 記号 2. 英数字 3. ひらがな 4. カタカナ 5. ギリシャ文字 6. ロシア文字 7. 罫線 8. 部首入力 0. 画数入力 a. 第一水準 b. 第二水準 c. 補助漢字

C-n	次の一覧部分を表示する
C-p	前の一覧部分を表示する

### 単語登録

M-x toroku-region	リージョン (矩形) 指定した単語を登録する
M-x edit-dict-item	単語を登録した辞書を編集する

### その他

x?	小さい文字を出す (例: 「xa」で「あ」)
z?	特殊記号を出す (例: 「z(」で「【」)

## z を用いた記号の入力

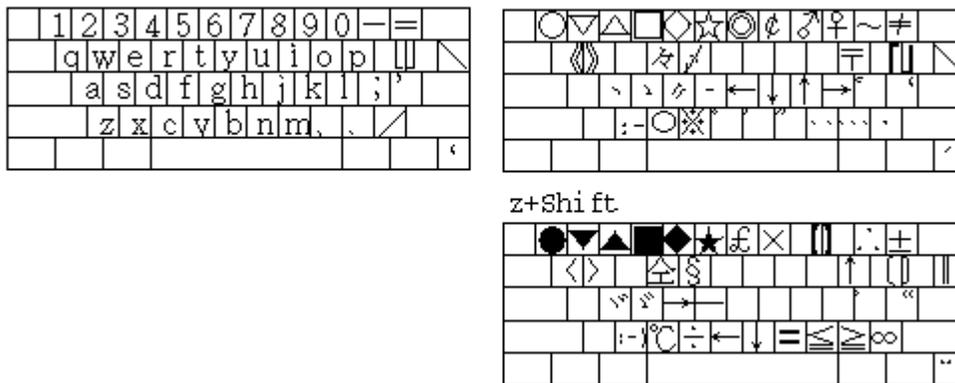


図 A.1 z キーとの組合せによる記号

## MHE

### 読む

M-x mh-rmail <return>	MHE メールリーダを起動する
q	MHE メールリーダを終了する
.	メールを読む
<Space>	続きの部分を読む
<Delete>	前の部分を読む

### 書く

M-x mh-smail <return>	メールを書く
C-c C-c	メールを送信する
C-c C-q	メールを送信するのをやめる

### 返事

a	メールに返事を書く
C-c C-y	メールの内容を引用する
C-c C-c	返事を送信する
C-c C-q	返事を送信するのをやめる

### 整理

o <i>foldername</i>	メールを~/Mail 以下のフォルダに振り分ける
---------------------	--------------------------

### メールボックス

M-f <i>folder_name</i>	~/Mail 以下のフォルダのメールを読む
M-r	現在のフォルダを読み直す。古いメールを読み返す時に便利。
M-p	メールの番号を振り直す

## GNUS

M-x gnus <return>	GNUS を起動する
q	GNUS を終了する
ニュースグループ選択画面	
<Space>	カーソル位置のニュースグループを読む
c	すべて既読にする
u	次の起動から表示しないようにする
L	すべてのニュースグループを表示する
記事画面	
<Space>	カーソル位置のネットニュースを読む
<Space>	記事の続きを読む
q	ニュースグループ選択画面に戻る
<Delete>	記事の前の部分を読む
d	記事に既読マークを付ける
f	表示中の記事に対してフォローする
F	表示中の記事に対して引用付きでフォローする
a	投稿のための原稿を編集する
C-c C-c	(フォロー、投稿記事編集画面で)記事を投稿する
o	記事を保存する

## Directory モード

M-x dired <return>	Dired モードを起動する
q	Dired モードを終了する
f	カーソル位置のファイルを読み込む。ディレクトリなら移動
v	カーソル位置のファイルを見る。元に戻るのは C-c
~	親ディレクトリに移動
~	バックアップファイル filename に削除マークを付ける
d	カーソル位置のファイルに削除マークを付ける
u	カーソル位置のファイルのマークを取り消す
x	マークファイル(削除など)を実行
C	カーソル位置のファイルのコピー
D	カーソル位置のファイルの削除
R	カーソル位置のファイルのリネーム
M	カーソル位置のファイルの chmod

## C モード

M-x c-mode	C モードにする
M- C-\ (または M-x indent-region)	リージョンの範囲をインデントする

## コンパイル

M-x compile <return>	コンパイラを起動する
----------------------	------------

## .emacs の設定例

```
(setq enable-double-n-syntax t)
```

「nn」で「ん」と変換するようにします。

```
(load "/NF/local/general/lib/mule/19.28/lisp/its/hira.el")
```

```
(its-defrule "string" "string2")
```

但し、emacs の場合は

```
(defrule "string" "string2")
```

string をローマ字入力すると string2 になるようにします。

例

```
(load "/NF/local/general/lib/mule/19.28/lisp/its/hira.el")
```

```
(its-defrule "dhi" "でい")
```

```
(its-defrule "thi" "てい")
```

```
(setq-default case-fold-search nil)
```

case-fold-search という検索時に大文字小文字を区別するかどうかのデフォルト値を設定します。この場合は全てのバッファにおいて区別します。

```
(global-set-key "\C-x@" 'compile)
```

C-x @のキー操作に対して compile のコマンド操作を割り当てます。因みに M- C-a なら "\e\C-a" と表します。

```
(autoload 'gnus "gnus" "Read Network News" t)
```

起動時に gnus 関数を自動的に読み込みます。

```
(setq kill-whole-line t)
```

通常、一行削除は行頭で C-k を 2 回行なう必要がありますがこれで一回で済むようになります。

## A.4 京都産業大学 FAQ(抄)

### A.4.1 目次

はじめに (372 ページから)

FAQってなんですか？

この FAQ リストの最新版 (全部) はどうしたら得られますか？

FAQ に載っていないトラブルなんです、どうすれば良いですか？

補助員ってなんですか。

UNIX 編

各種のコマンドに関する事 (372 ページから)

コマンドの実行結果をプリントアウトしたいのですが？

grep の使い方、正規表現がよく分かりません。

実行中のプロセスの終了

LINKって何？

テキストのファイルがぐちゃぐちゃで読めません

Mule に関する事 (375 ページから)

mule でのかな漢字変換で「thi」で「てい」と変換するにはどうすれば良いのですか？

Mule で単語登録したものの一覧は得られますか？

メールに関する事 (376 ページから)

メールを相手が読んだかどうかを確認したいのですがどうすればよいですか？

シグネチャを付けるにはどうすれば良いのでしょうか？

相手のメールアドレスが判らないのですが、調べる方法はありますか？

特定の人から来たメールだけ別のフォルダに入れる方法はありませんか？

ネットニュースやメールの返事を書く時に引用符が付けられなくなっちゃいました。

フォルダ内のメールの番号を日付順にするにはどうしたらいいのですか？

ネットニュースに関する事 (377 ページから)

ニュースグループのソートの仕方 (GNUS)

ネットニュースを読んでいたら、ニュースグループの頭に「\*」が付いてしまいました。偶然の産物なので、消し方を知りません。どうすれば消えるのでしょうか。

時々相手が文頭や文末に自分のことを「尾崎@計算機センターです」などというように書いていますが、これはどういう意味ですか？

シグネチャを付けるにはどうすれば良いのでしょうか？

ネットニュースに記事を投稿したのですが、うまく投稿できたかどうかを確認するには、どうすれば良いのですか？

いろんなニュースグループがありますが、それぞれどんなものなのですか？

クロスポストってなんですか？

Followup-To: ってなんですか？

X に関する事 (380 ページから)

リモートログインした機械でアプリケーションを立ち上げようとすると Can't open display が出て出ます。

それ以外のこと (381 ページから)

パーミッションって何ですか？

ディレクトリを他の人からも見られるようにしたいんですが

キーボードを打っても文字が化ける、あるいは何も表示されずとも動かないのですが

印刷したいのですが、どのプリンタを指定したら良いのでしょうか？

家のパソコンとデータをやりとりしたいのですが

Mac 編 (382 ページから)

マックにリセットスイッチはついていないのですか？

Mac でフロッピーが取り出せない。

Mac でディスクがロックされている。ディスクに保存できない。

Mac でことえり入力時にカタカナしか出ない。

Program 編 (382 ページから)

math.h を使ったらコンパイルできない。

その他 (382 ページから)

フロッピーディスクを買いたいのですが、

## A.4.2 はじめに

Q. FAQってなんですか？

A. 何度も繰り返される質問をまとめた Q & A 集です。

Frequency asked question の略です。何度も同じ質問が出ると答える方も疲れるし、質問する方も気が引けてくるので、まずこれを見ることでそれを解決しようという意図があります。ここでは学内にあ  
る計算機的环境に対するさまざまな質問に答えています。

Q. この FAQ リスト (の最新版) はどうしたら得られますか？

A. 次の内からお選びください。

- ・ 計算機センターメインカウンター ( 計算機運用補助員待機場所 ) まで取りに行く。
- ・ World Wide Web の各種ドキュメントにある F A Q を見る。
- ・ sandai.question に定期的に投稿されるものを見る。

Q. 補助員、または MiCS 補助員ってなんですか？

A. アルバイトの学生によるコンピュータ環境のお助け部隊です。

皆さんは学内のコンピュータ施設を使っていて、何か困った事があったことはないでしょうか。プリンタから印刷されてこない、コンピュータが使用中に止まってしまった、このアプリケーションの使い方が分からない、等々。そんなときにはぜひ補助員を呼んでください。

補助員とは正確には「計算機運用補助員」という名前で、計算機センターで学内のコンピュータのトラブルに対応するために待機している、その方面の知識を持った学生達のアルバイトです。「計算機運用補助員」ではイメージが固いので MiCS 補助員というニックネームを付けました。以下 MiCS 補助員と呼びます。

MiCS 補助員は学内で授業が行われている間中、複数の人間が勤務しており、少なくとも一人は計算機センターが管理している 11, 21, 32, 51 情報処理教室、それぞれの部屋を何かトラブルはないかと巡回を続けています。また少なくとも一人は計算機センターで常時待機しており、いつでもコンピュータ施設のトラブルに対する電話を受け付けています。あなたがもしどこかの情報処理教室でトラブルに遭遇した時、その部屋にタイミング良く補助員がいれば解決しますし、いなくても、各情報処理教室に設置されている内線電話を使って補助員に連絡すれば電話で、もしくはかけつけて問題を解決してもらえというわけです。

MiCS 補助員を呼ぶには先ほども書いたように各情報処理教室に設置されている内線電話を使って「2578」をプッシュしてください。すると「MiCS メインカウンター」というところにつながり、補助員がトラブルの状況を聞いてきます。もしそれが簡単な対処法で解決するような問題ならば電話で補助員が解決方法の指示を出します。そうでなければ補助員が何らかの方法で解決しますので指示を聞いてください。

また補助員は年度の初めに新たにスタッフの募集をしております。コンピュータをやってみようという意欲のある方、応募をお待ちしています！

- MiCS 補助員 -

業務内容：計算機センターが管理しているコンピュータ施設のトラブル全般の解決

業務期間：学内で通常講義のある期間全て

特典：計算機に対する知識が得られる

連絡方法：各情報処理教室に設置してあるコ - ドレスホンで 2578 をプッシュしてください

## A.4.3 UNIX 編

各種のコマンドに関すること

Q. FAQ に載っていないトラブルなんですが、どうすれば良いですか？

A. 補助員やよく知っている人に聞きましょう。

計算機について分からないことがあれば、各情報処理教室にあるコードレス電話で 2578 をダイヤルしてください。MiCS 補助員というナイスガイ&ナイスギャル達がお答えします。また、お急ぎでない場合や、専門的な質問は京都産業大学ローカルニュースグループの sandai.question に投稿してください。誰か知っておられる方からフォローが入るでしょう。

Q. コマンドの実行結果をプリントアウトしたいのですが？

A. いくつか方法があります。

まず script コマンドを利用する方法。例えば、

```
cc2000(121)% script filename
```

とすると exit を実行するまでのことが filename という file に書き込まれます。これは画面に出力される事がそのまま書き込まれます。

もう一つ、リダイレクトを利用する方法。

```
cc2000(127)% a.out > filename
```

とすると実行結果が filename という file に出力されます。

他には、既に実行されて画面に結果が表示されているならば、それを Cut&Paste で

```
cat > filename
```

を使って filename のファイルにする事もできます。

後は lpr コマンドでプリンターに出力してください。

Q. grep の使い方、正規表現がよく分かりません。

A. grep はある文字列を含む行の一覧を出力するコマンドです。たとえば、

```
% grep hello sample.txt
```

または

```
% cat sample.txt | grep hellow
```

とすれば、sample.txt ファイル中の”hello”という文字列が出現する行が全部画面（正確には標準出力）に出てきます。

文字列にはワイルドカード（「\*」や「?」）も使用可能です。

サポートしている正規表現は

^ : 行頭に Match

\$ : 行末に Match

. : CR (改行 Code) 以外の任意の 1 文字に Match、2 Byte Code も Document の Script に応じ 1 文字として認識

です。このように使います。

例：  
1: sample document 12345  
2: 12345  
3: 123456789 this is a test  
4: abcdefg9

grep '^123'	で、2: 3: がヒット	(先頭から 123 があるもの)
grep '123'	1: 2: 3:	(123 があるもの)
grep '9\$'	4:	(9 で終るもの)
grep '9'	3: 4:	(9 を含むもの)
grep 'a...e'	1: 4:	(a に続いてどんな文字でもいいから 3 文字あって、e があるもの)
grep 'ae'	全くヒットしない	(ae があるもの)
grep 'a*e'	全くヒットしない	(ae, aae, aaaa, ... を含むもの)
grep 'a.*e'	1: 3: 4:	(a に続いてどんな文字でもいいから 何文字かあって (ゼロ文字でも良い) e があるもの)

Q. 実行中のプロセスの終了

A. ps コマンドと kill コマンドを使用します。

とりあえず ps コマンドを実行します。すると以下のようなものができます。

```
cc2000(82)% ps
  PID TT      S   TIME COMMAND
 25830 pts/10  S   0:00 -tcsh
 25138 pts/12  0   0:00 ps
 29506 pts/12  S   0:01 -tcsh
 25931 pts/13  S   0:00 -tcsh
 26171 pts/13  S   0:41 mule
```

これで目的とする物がでてこないなら次のようにします。但しこれは cc2000 の場合ですのでそれ以外の機種は下記の表に対応するオプションを指定してください。

```
-----
ホスト名      O S      オプション      備考
cc2000        Solaris2  -axu             /usr/ucb/ps の場合
               /bin/ps の場合は -ef
cssol????    Solaris2  -aux             cc2000 と同じ
axt?????     Linux     axu
--            標準的な SVR4  -ef
--            標準的な BSD   -axu             SunOS なら -xu で良い
-----
```

```
cc2000(83)% ps -aux | grep ozaki
ozaki  24908  0.2  0.1  972  836 pts/12  0 14:06:57  0:00 ps -aux
ozaki  24909  0.1  0.1  708  484 pts/12  S 14:06:57  0:00 grep ozaki
ozaki  29506  0.1  0.2 1080  996 pts/12  S 09:11:17  0:01 -tcsh
ozaki  25830  0.0  0.2 1084  948 pts/10  S 08:30:45  0:00 -tcsh
ozaki  25931  0.0  0.2 1080  940 pts/13  S 08:31:09  0:00 -tcsh
ozaki  26171  0.0  0.5 4472 3280 pts/13  S 08:31:19  0:41 mule
ozaki  26196  0.0  0.2 1080  952 pts/14  S 08:31:52  0:01 -tcsh
```

このようにすると全てのプロセスから ozaki さんのプロセスを表示します。パイプより前の部分でシステム中の全プロセスをリストして、パイプより後ろの部分で前半の結果から自分のユ - ザ名 (例では ozaki) の文字列を含む行だけ抜き出して表示させている訳です。これで自分のプロセスが表示されます。ですから上記の ozaki を自分のユ - ザ名に置き換えると自分のプロセスを表示してくれます。

そして一番左の数字がプロセス番号ですので、殺したい (終了させたい) プロセス番号 (PID) を指定した kill コマンドを実行します。

```
cc2000(83)% kill 26171
```

これで、もし止まらない場合はもう少し強力的に止めるオプションをつけます。

```
cc2000(83)% kill -1 26171
```

単なる kill (実は kill -15 と同じ), kill -1, kill -2, kill -9 の順に強力的になりますので、順に試してください。

## Q. LINKって何?

### A. ファイルを扱う方法の一つです。

LINK にはシンボリックリンクとハードリンクという2種類が存在します。ファイルというのはシステム的に見ると、2つに分割して考えることができます。

- ・ 現実にディスクに書き込んである内容そのもの。
- ・ その開始位置を指し示しているポインタ。

普段私たちが目にしているファイル名は2つめのポインタにあたります。

シンボリックリンクとはファイル名を指すファイル名です。下記のように file1 を指す file2 のようなものです。file2 を参照すると、file1 を見にいき、それは file の内容を指しているので無事 file の内容を参照できる訳です。

File の内容

```
file1  file2
```

ハードリンクとはファイルの内容を指し示すもう一つの名前をつけてやる事をいいます。

## File の内容

file1 file2

通常使うのはシンボリックリンクの方です。リンクの利点は主に二つあります。一つはディレクトリ構造の離れた所にあるファイルを指定してアクセスし易くする事。もう一つはコピーと違ってファイルの実態は一つなのでディスク容量を節約することができることです。ハードリンクは普通ディレクトリに対して使うことが多いです。例えば、プログラムのコンパイル等でライブラリの指定をリンクで行ってれば、それを変える事でライブラリを簡単に切り換える事もできます。

### シンボリックリンクの設定方法

```
% ln -s file1 file2
```

ここで file1: 元のオリジナルファイル名

file2: 元ファイル名 (file1) を指し示すようにしたいファイル名

この時に file1 を消すと file2 は指し示すファイルが無くなるので名前だけ残って使えないファイルになってしまいます。

File の本体	file1 を消す	File の本体
file1 file2		??? file2

それに対してハードリンクは

```
% ln file1 file2
```

と実行します。この場合、file1 と file2 の立場は同等で、実体は同じで名前を二つ持っている状態になります。この時は file1、file2 のどちらかを消してももう一つの側でファイルにアクセスすることができます。

File の本体	file1 を消す	File の本体
file1 file2		file2

Q. テキストのファイルがぐちゃぐちゃで読めません

A. ファイルの文字コードを変換します。

漢字やひらがな等のアルファベット 2 文字分の大きさのある文字 ( 全角文字 ) は規格によりその文字のコード ( 文字を記号とみなした通し番号 ) が違うので違う規格で読むと全然違う文字が表示されてしまいます。主に本学では EUC コードを用いていますがメールやネットニュースでは JIS コードも用いられています。cc 環境では nkf という Network Kanji code conversion Filter があります nkf コマンドを使ったコード変換は次のようにして行ないます。

1. ファイルを一旦別の名前にして、
2. nkf で 目的のコードに変換し、それを 元のファイル名に書き出し、
3. 別の名前にしておいたファイルを消去する

というところです。以下に具体的な手続きを。

1. cc2000(20)% mv filename1 filename2
2. 目的に合わせて次の何れかを選んでください。  
それぞれ、j(JIS) e(EUC) s(ShiftJIS) に変換します。  
cc2000(21)% nkf -j filename2 > filename1  
cc2000(21)% nkf -e filename2 > filename1  
cc2000(21)% nkf -s filename2 > filename1
3. cc2000(22)% rm filename2

## Mule に関すること

Q. mule でのかな漢字変換でたとえば「thi」で「てい」と変換するにはどうすれば良いのですか？

A. .emacs を編集します。

まず

```
cc2000(01)% mule .emacs
```

としてください。これで.emacs という mule の環境設定ファイルを読み込み mule が起動します。今まで何も変更していない人は

```
;;  
;;      Emacs common settings.  
;;  
  
(load "/NF/home/common/settings/_emacs.load")
```

このような内容になっていると思います。

ここで、(load... の行の次の行に

```
(load "/NF/local/general/lib/mule/19.28/lisp/its/hira.el")  
(its-defrule "thi" "てい")  
(its-defrule "cya" "ちゃ")  
(its-defrule "cyi" "ちい")  
(its-defrule "cyu" "ちゅ")  
(its-defrule "cye" "ちえ")  
(its-defrule "cyo" "ちょ")
```

のように書いて保存してください。すると次に mule を起動した時から反映されます。

## メールに関すること

Q. メールを相手を読んだかどうかを確認したいのですがどうすればよいですか？

A. 自動的に相手を読んだら確認できるようなシステムはありません。

もし必要ならメールに読んだら返事をしてくれるよう書いておけば良いでしょう。  
また、NeXT メールなら受領証という機能がありますのでそれで確認することができます。

Q. シグネチャを付けるにはどうすれば良いのでしょうか？

A. あらかじめファイルを作っておいてそれを挿入します。

前もってシグネチャをファイルに書いて作っておきます。そして必要に応じてその内容を取り込みます。そうすると毎回タイプする必要がなくなりますね。

mule を使ってメールを書いている人を対象に説明します。例えば sig というファイル名で 1 ~ 4 行程度のシグネチャを作っておきます。そしてメールを書き終わった時に C-x i と (コントロールキーを押しながら x キーを押して、コントロールキーを離して i キーを押す) すると、下に Insert file: ~/ の様なメッセージが出ると思われます。出たら、そのメッセージに続けて sig とタイプします。

```
Insert file: ~/sig
```

となりましたか？ なかったらリターンキーを押してください。無事にシグネチャがカーソルのあった位置へ書き込まれたと思います。なお、余り長いシグネチャはマナーに反しますので 4 行までにしましょう。

Q. 相手のメールアドレスが判らないのですが、調べる方法がありますか？

A. 基本的に調べる方法はありません。

ただし cc 環境ならば whois コマンドを利用して調べる方法があります。

Q. 特定の人から来たメールだけ別のフォルダに入れる方法はありませんか？

A. 読み終えたメールを振り分けることができます。

mule でメールを読んでいる時に o を押すと

```
Destination folder? +
```

と聞いてきます。ここで、そのメールを保存したいフォルダを入力して<return>を押すと振り分けてくれます。指定したディレクトリが存在しない場合は

```
Folder +hoge hoge does not exist. Create it? (y or n)
```

と聞いてきますので y と答えてください。

まとめて行ないたい場合は、コマンドで行ないます。例として、suzu3 から来たメールをフォルダ suzu3 に入れる場合

```
cc2000(81)% refile 'pick -from suzu3' +suzu3
( ' と ' を間違えないように! )
```

これで ~/Mail/inbox のメールのうち、suzu3 から来たメールが全て ~/Mail/suzu3 に移されます。~/Mail/suzu3 が存在しなければ

```
Create folder "/NF/home/syokuin0/ozaki/Mail/suzu3"?
```

と聞いてきますので y と答えれば OK です。

Q. ネットニュースやメールの返事を書く時に引用符が付けられなくなっちゃいました。

A. supercite がうまく動いていないのです。

具体的にはメールを読んだ後、それを引用してメールを書こうとして C-c C-y とすると

```
Wrong type argument: listp, " . "
```

と表示される場合です。

ホームディレクトリ以下に、.src.el というファイルがあると思いますが、これが supersite という引用符号を付けるプログラムが使う設定ファイルです。

今まで定義した引用記号の設定が失われてもいいのであれば、このファイルを削除してやればうまく行くようになると思います。

設定が失われるのが嫌であれば、このファイルの中を覗いて、怪しげな部分を手で修正する (mule で開いて修正、保存する) というのが良いと思います。

Q. フォルダ内のメールの番号を日付順にするにはどうしたらいいのですか？

A. mule + mh-e の機能を使います。

mule or emacs + mh-e を使ってるのでしたら、そのフォルダを visit している状態 (メールの subject の一覧がでている状態) で

```
M-x mh-sort-folder <return>
```

をすると日付順に番号をつけ直してくれると思います。

mule や emacs で何かする機能が無いかどうかを調べるには「並びかえる」は「sort」なので C-h a sort <return> とすると「sort」という文字列を含んだ関数の一覧が表示されます。その一覧の中で「mh」や「folder」という文字列を含むものを探すと「mh-sort-folder」を見つけることができます。

## ネットニュースに関すること

Q. ニュースグループのソートの仕方 (GNUS)

A. 直接順番を入れ換えるか、ファイルをソートします。

ニュースグループ選択画面のニュースグループを編集して行を入れ変えるとその順番になります。

```
:
9: sandai.comp.announce
4: sandai.comp
1: sandai.general
11: sandai.rec
:
```

例えばこのようなニュースグループの並びになっていて、sandai.rec を sandai.comp の上に持ってきたい、というような場合は、まずカーソルを sandai.rec の行の先頭 (1 カラム目) に移動させます。ここで C-k (コントロールキーを押しながら k) を二回実行すると sandai.rec の行が消えます。消えたら今度は sandai.comp の行の先頭にカーソルを移動させて C-y を実行します。

```
:
9: sandai.comp.announce
11: sandai.rec
4: sandai.comp
1: sandai.general
:
```

こうなりましたね。これで今後もこの順序でニュースグループが表示されます。  
他にはコマンドラインからソートする方法もあります。

```
sort <元ファイル名> -o <出力ファイル名>
```

でアルファベット順にソートできます。そしてニュースグループが書いてあるファイルは~/newsrcc-cc2000  
なので、自分のホームディレクトリで

```
cc2000(20)% sort .newsrcc-cc2000 -o .newsrcc-cc2000
```

というコマンドを実行してやればニュースグループをソートすることができます。

!!! やってはいけないこと !!!

```
cc2000(20)% sort .newsrcc-cc2000 > .newsrcc-cc2000
```

こういう書き方は一見できそうな気がするのですが、> .newsrcc-cc2000 の部分で書き込み場所を先に確保してしまうので .newsrcc-cc2000 が初期化され無くなってしまいます。無くなった物をソートしても何も残るはずは無く、.newsrcc-cc2000 というファイルは中身なしのファイルになってしまいます。

Q. ネットニュースを読んでいたら、ニュースグループの頭に「\*」が付いてしまいました。偶然の産物なので、消し方を知りません。どうすれば消えるのでしょうか。

A. gnus で読んでいる時に、u を押してしまったのではないのでしょうか？

記事を読んでいる時に u を押すと、mule の上の画面のその NG の記事一覧の一番左にあるマークが - になって、既読で記事を消去するのを一時保留します。(普通は、ここが D になって q で抜けると、D マークがついた記事は既読として消去されてしまいますよね。)それで、NG 一覧の画面では、そういうファイルが NG の中にあると、その横に\*をつけて、以上の処理をした記事がありますよと、知らせます。だから、もういちどその記事を読んだことにするために記事の場所で'd'を押すと既読マークがつくので、すべて解決となるはずですよ。

Q. 時々相手が文頭や文末に自分のことを「尾崎@計算機センターです」などというように書いていますが、これはどういう意味ですか？

A. これは「尾崎」さんが「計算機センター」の人ですという意味です。

要は「@」マークが区切りとなっていて、個人@所属を表します。こういう書き方の元はメールアドレスからきています。メールアドレスは honyarara@cc.kyoto-su.ac.jp という風になっていますね。これは honyarara さんが jp(日本)の ac(大学・研究機関)の kyoto-su(京都産業大学)の cc(ホストマシン名)の人ですという意味です。きちんと個人@所属になっていますね。

尚、「個人%コメント等@所属」という表記もあるようです。これにも由来はあるのですがここではふれません。参考程度に覚えておいてもいいでしょう。

Q. シグネチャを付けるにはどうすれば良いのでしょうか？

A. あらかじめファイルを作っておいてそれを挿入します。

前もってシグネチャをファイルに書いて作っておきます。そして必要に応じてその内容を取り込みます。そうすると毎回タイプする必要がなくなりますね。

mule を使ってメールを書いている人を対象に説明します。例えば sig というファイル名で1~4行程度のシグネチャを作っておきます。そしてメールを書き終わった時に C-x i と(コントロールキーを押しながら x キーを押して、コントロールキーを離して i キーを押す)すると、下に Insert file: ~/ の様なメッセージが出ると思われます。出たら、そのメッセージに続けて sig とタイプします。

```
Insert file: ~/sig
```

となりましたか？ なかったらリターンキーを押してください。無事にシグネチャがカーソルのあった位置へ書き込まれたと思います。なお、余り長いシグネチャはマナーに反しますので4行までにしましょう。

Q. ネットニュースに記事を投稿したのですが、うまく投稿できたかどうかを確認するには、どうすれば良いですか？

A. その記事を自分で読みましょう。

GNUS を使っているならニュースグループ選択画面で g キーを押す、または GNUS を一度終了して、もう一度起動すると新しく投稿された記事を読み込むのでそれで確認してください。ただし投稿後しばらくしないと記事が処理されません(最大15分)ので注意してください。

Q. いろんなニュースグループがありますが、それぞれどんなものなのですか？

A. 以下に一覧を挙げておきます。

京都産業大学に限ったニュースグループ

sandai 京都産業大学からのみ読み書きできるニュースグループ。まずはここで慣れてから fj.\* 等を読み書きすると良いでしょう。

京都産業大学以外の一般的なニュースグループ

fj 主に日本語によって論議をするためのもの。特定の組織や団体が管理・運営をしている訳ではなく、利用者全員の合意に基づいて運営されている。現在のところ営利目的の記事の投稿はできない。

tnn IJ が主催する営利目的の記事を投稿できるニュースグループ

jp 日本の IP 接続組織用のニュースグループ

comp USENET のうちコンピュータに関する話題用

news USENET のうち NetNews に関する話題用

rec USENET のうち趣味に関する話題用

sci USENET のうち科学に関する話題用

soc USENET のうち社会問題に関する話題用

talk USENET のうち各種の論議用

misc 上記以外の USENET の話題用

alt USENET ニュースグループ群に代わる、各種の話題用。非常に緩いルールしかなく、かなり自由にニュースグループを作成して利用できる。

bionet 医学・生物学の話題用

biz ビジネスの話題用

gnu FSF(フリーソフトウェアファウンデーション) やその製品 (GNU プロダクト) に関する話題用

k12 アメリカにおけるネットワークを使った教育の実験のためのニュースグループ

vmsnet DEC VAX/VMS OS に関する話題用

Q. クロスポストってなんですか？

A. 複数の NG に同じ記事を投稿する時に使用する機能です。

クロスポストとはネットニュースの記事をポストする時、ポスト先のニュースグループを複数指定 (通常 2 つか 3 つ) して、同じ記事を各ニュースグループに投稿するものです。各ニュースグループに個別に投稿するのに比べて、記事の実体は一つで済みますので、ネットワーク資源の節約になります。

方法は、通常のポスト時に Newsgroup の指定行を

```
Newsgroups: sandai.test, sandai.junk
```

のようにカンマで区切って複数のニュースグループを書きます。この状態で投稿すると指定した各ニュースグループに投稿します。この状態の記事にフォローを入れた場合、Newsgroups の指定はそのまま受け継がれるので全てのニュースグループに同じフォローを入れる事になります。(このフォロー先を一意に決めたい時はフォローアップを使います。)

通常クロスポストを使うのは何箇所かのニュースグループで同時に議論したいときと、そのニュースグループで議論を続けるのにふさわしくない記事に対して、他のニュースグループにふる時につかいます。前者の場合は議題が幾つかのニュースグループにまたがっていて、どうしてもそれら全てのニュースグループの読者の意見を交えて議論したい、という場合です。通常のポストの際に Newsgroups に上記のように二つ以上のニュースグループを指定してください。

後者の場合は、フォローして、そのニュースグループと新しいニュースグループとのクロスポストにして、フォローアップを新しいニュースグループにします。例えば sandai.question のある記事に対して sandai.junk の方が適切だ、と思ったらフォローする際に次のようにします。まず F キーを押して引用符を決めるとこの様になりますね。

- - - ここから - - -

```
In-reply-to: ozaki@cc.kyoto-su.ac.jp's message of 26 Jan 1995 12:39:48 +0900
```

```
Newsgroups: sandai.question
```

```
Subject: Re: .emacs
```

```
Distribution: local
```

```
References: <TANIMA.95Jan26114834@cssol020.kyoto-su.ac.jp>
```

```
<OZAKI.95Jan26123945@cc2000.kyoto-su.ac.jp>
```

```
--text follows this line--
```

(引用した本文)  
- - - ここまで - - -

通常、「-text follows this line-」より上の部分は書き換えないのですが、Newsgroups: 行の変更と、Followup-To: 行の追加を行ないます。

Newsgroups: sandai.question, sandai.junk  
Followup-To: sandai.junk

この様に修正、追加します。次のようになりましてね。

- - - ここから - - -  
In-reply-to: ozaki@cc.kyoto-su.ac.jp's message of 26 Jan 1995 12:39:48 +0900  
Newsgroups: sandai.question, sandai.junk  
Followup-To: sandai.junk  
Subject: Re: .emacs  
Distribution: local  
References: <TANIMA.95Jan26114834@cssol020.kyoto-su.ac.jp>  
          <OZAKI.95Jan26123945@cc2000.kyoto-su.ac.jp>  
--text follows this line--  
(引用した本文)  
- - - ここまで - - -

二行以外は変更ありません。これでいつものようにポストするとその記事は sandai.question と sandai.junk に投稿され、それぞれの記事を見た人がそれにフォローしようとするとならぬグループに sandai.junk が選択されます。

そして、一言、「sandai.junk に振ります」と書き添えてください。こうすると sandai.question の読者は sandai.junk に議論の場が移るんだ、ということが分かりますし、sandai.junk の読者は sandai.question から移ってきた議論だ、ということが分かります。

**Q. Followup-To: ってなんですか？**

**A. 記事に対してのフォロー先のニュースグループを決めるものです。**

例えば sandai.test にある記事に

Followup-To: sandai.junk

となっていたらその記事のフォローは sandai.junk に投稿されます。

## X に関すること

**Q. リモートログインした機械でアプリケーションを立ち上げようとする Can't open display が出てきます。**

**A. 貴方が使っている機械の環境を変えなければいけません。**

これはそのリモートログインしている機械が貴方の使っている機械に表示しようとしたけれどもできませんでした、という事です。Xウィンドウはネットワークを介して繋がっている機械に画面を表示する為に作られたウィンドウシステムで、その為、何処に表示するのか、何処から自分の画面への表示を許すのか、を指定してやる必要があります。Can't open display とは表示しようとしている機械に表示できませんでした、ということです。他からの画面の書き込みを許すには xhost というコマンドを使います。

% xhost 機械名

とするとその機械からの表示を許可します。何処に表示するかは環境変数の DISPLAY で指定するのでログインした先のコマンドラインで

% setenv DISPLAY 自分の使っている端末の名前:0.0

と入力します。これで、目の前の端末が、ログイン先の機械の画面を表示できるようになります。

例

```
cssol011(81)% xhost cc2000
```

```
cc2000 being added to access control list
```

```
cssol011(82)% rlogin cc2000
```

Last login: Thu Sep 22 11:00:56 from cssol031  
tcsh: using dumb terminal settings.  
Sun Microsystems Inc. SunOS 5.3 Generic September 1993

cc2000(81)% setenv DISPLAY cssol011:0.0

(下線部が入力部分)

setenv の 0:0 は 0 番目のディスプレイの 0 番目の画面という意味で、この意味のよく解らない人はおまじないとして打てば O.K.

それ以外のこと

Q. パーミッションって何ですか？

A. ファイルに対するアクセス権の事です。

パーミッションとはファイルに与えられるアクセス権で、オーナー、グループ、他人の三種類の人に対してリード権、ライト権、実行権があるか無いかで決めます。詳しくはコンピュータガイド インターネット編「UNIX もっともっと ファイルのアクセス権」を見てください。

Q. ディレクトリを他の人からも見られるようにしたいんですが

A. ディレクトリのアクセス権を変更します。

そのディレクトリに対するアクセス権を変えます。読めるようにするにはリード権と実行権を与えます。さらに書き込めるようにするにはライト権を与えます。あるグループに対して解放するなら所属グループを変更します。

この文章を読んで意味が分からない人はセキュリティ上、危険ですので少し勉強した方が良いと思われれます。例えば自分宛のメールを覗かれたり大切なデータをいじられたりする危険性があります。自信のない人は分かっている人に見て貰ってください。

Q. キーボードを打っても文字が化ける、あるいは何も表示されずまともに動かないのですが

A. 状況として3通り考えられます。

一つは何かバイナリデータなどの普通表示できないデータを画面に表示してしまった場合、端末設定が変わってしまい、以後の文字がすべて化けてしまう場合。そしてカナキーや CAPS キーなどの特殊なキーを押してしまった場合。もう一つは C-s(コントロールキーを押しながら s キーを押す)の場合です。順に対処法を。

一番目の場合

- ・端末状態をリセットする
- ・端末エミュレータを再起動してやる

二番目の場合

カナキーや CAPS キーを解除する。(もう一度押す)

三番目の場合

C-q (コントロールキーを押しながら q キーを押す)をする。

Q. 印刷したいのですが、どのプリンタを指定したら良いのでしょうか？

A. ccinfo コマンドで参照できます。

% ccinfo<return>で ccinfo コマンドを起動して、「cc 環境の設備について」の「プリンタの配置について」を見てください。

Q. 家のパソコンとデータをやりとりしたいのですが

A. コンピュータガイド -インターネット編- FTP の節をお読みください。

#### A.4.4 Mac 編

Q. マックにリセットスイッチはついていないのですか？

A. キーボードによるリセットがあります。

ctrl とアップルキー (花文字の奴) を押しながら電源を入れる時に押す、キーボードの上にある四角いボタンを押します。(一部機種では使えません)

Q. Mac でフロッピーが取り出せない。

A. フロッピーに保存された書類からアプリケーションを立ちあげていませんか？

作業中に書類を保存して、一旦アプリケーションを終了させてください。取り出せるようになります。もし、フロッピーが入ったままハングしたときは、アップル+シフトキー+数字の 1 を試してください。それで駄目なら、マウスのボタンを押したままリブートしてください。大概はこれで吐き出されるはずですが。

ちなみに、この 2 つは Macintosh での、フロッピーディスクの強制 eject 方法です。もし、これでも駄目なら、もう一度、アップル+シフトキー+数字の 1 を押してください。

どうしても無理なら MiCS 相談室 (内線 2578) までお電話ください。補助員が対処しに行きます。

Q. Mac でフロッピーディスクがロックされている。フロッピーディスクに保存できない。

A. フロッピーディスクが書き込み禁止状態になっていませんか？

フロッピーディスク (差し込む方向に持って、そのまま手を返してください。向かって右側にあります) のスライド式の黒い小さなノブ (ライトプロテクトノッチ) を書き込み禁止状態 (穴が空いている状態) にしていませんか？

一旦フロッピーを取り出して、確認してください。もしそうになっていなくとも、これで直る場合もあります。もう一度お試しください。

どうしても無理なら、MiCS 相談室 (内線 2578) までお電話ください。補助員が対処しに行きます。

Q. Mac でことえり入力時にカタカナしか出ない。

A. キーボード左下の Capslock キーが押されています。

もう一度 CapsRock キーを押して、上に飛び出た状態 (押し込まれていない状態) にしてください。それでも直らなければ、鉛筆メニューから「操作パレット表示」を選んで、操作パレットの「あ」という文字のボタンを押してください。シフトキーを押しながら打つと、カタカナが出ます。

#### A.4.5 Program 編

Q. math.h を使ったらコンパイルできない。

A. UNIX 上で C 言語で、sin 関数などのいわゆる math.h を include するようなプログラムを cc しようとした時 NeXT ではそのままコンパイルできるのですが、cc2000 や cssol や SUN では数学ライブラリを組み込む -lm オプションが必要です。

例: `cc test.c -lm`

# それでもできない時... プログラムミスかな? (^\_^;)

#### A.4.6 その他

Q. フロッピーディスクを買いたいのですが、

A. 基本的に電気屋さんやパソコンショップで売っています。種類は 3.5' 2HD (サンテンゴインチ ニイイチディー) と呼ばれるものです。

学内では丸善などで取り扱っています。但し少し市場より値が高いので大量に買うのなら他のパソコンショップなどで買う方が良いでしょう。また、異なる機種間でデータのやり取りをされるのであれば 3.5' 2DD (ニイディーディー) と呼ばれる種類の方がいい場合もあります。

# 付録B章 情報処理教室の利用について

学生便覧より抜粋

計算機を利用しての授業や自習のため、学内に情報処理教室を設置しています。情報処理教室は、次の3つに分類されます。

1. 10号館を中核とした全学共通の情報教育に使用される教室
2. 各号館に設置された10号館のサテライトとして主に学生の自習用に使用される教室
3. 各号館に設置された学部・学科の専門教育に高度に特化し学部・学科で運用管理される教室

これらの情報処理教室に設置している機器は、UNIX サーバの端末として利用できる他、パーソナル・コンピュータ（パソコン）としても利用できます。利用できるソフトウェアは各情報処理教室ごとに違いますので、その教室を管理する所管の事務室で確認してください。

#### 1. 情報処理教室の一覧

建物	教室名	OS 台数（予備）	所管
<b>10号館情報処理教室</b>			
10号館 2階	10201 情報処理教室	WindowsNT/Linux 89台 (5)	計算機センター
10号館 2階	10202 情報処理教室	WindowsNT/Linux 104台 (5)	計算機センター
10号館 2階	10203 情報処理教室	WindowsNT/Linux 47台 (2)	計算機センター
10号館 2階	10204 情報処理教室	WindowsNT/Linux 47台 (2)	計算機センター
10号館 2階	10205 情報処理教室	WindowsNT/Linux 47台 (2)	計算機センター
10号館 2階	10206 情報処理教室	Macintosh 47台 (2)	計算機センター
10号館 3階	10301 情報処理教室	WindowsNT/Linux 61台 (3)	計算機センター
10号館 3階	10302 情報処理教室	WindowsNT/Linux 89台 (5)	計算機センター
10号館 3階	10303 情報処理教室	WindowsNT/Linux 21台	計算機センター
1号館 2階	11 情報処理教室	Macintosh 91台	計算機センター
<b>サテライト教室</b>			
2号館 4階	21 情報処理教室	Solaris 44台	計算機センター
3号館 1階	32 情報処理教室	Macintosh 91台 (6)	計算機センター
5号館 1階	51 情報処理教室	WindowsNT 35台	計算機センター
<b>学部・学科の情報処理教室</b>			
計算機科学研究所棟 2階	C1 情報処理教室	WindowsNT/Linux 20台	計算機科学研究所
計算機科学研究所棟 3階	C2 情報処理教室	Solaris 20台	理学部
計算機科学研究所棟 3階	C3 情報処理教室	WindowsNT/Linux 15台	計算機科学研究所
計算機科学研究所棟 4階	C4 情報処理教室	NEXTSTEP 6台	理学部
3号館 2階	31 情報処理教室	Macintosh6100 44台	外国語学部
5号館 2階	52 情報処理教室	WindowsNT/Linux 39台	経営学部

2. 使用可能ソフトウェア一覧 10号館情報処理教室およびサテライト教室でのソフトウェア一覧です。学部・学科の情報処理教室については、所管の事務室にお問い合わせください。

10201～10205, 10301～10302 情報処理教室 (10号館)			
WindowsNT			
統合ソフトウェア	MicrosoftOfficePro	Excel 多変量解析	マクロ
ワープロ	Word	解析・シミュレーション	Win-RATS
表計算	Excel	ネットワーク関連	
資料提示	PowerPoint	Telnet	TeraTermPro
データベース	Access	WWW	Netscape
キーボード練習		FTP	WS-FTP95LE
統計解析プロダクト	SAS	画像処理	
	BASE,STAT,GRAPH,ASSIST,ETS		l-view32,D-Pixed,Susie
Linux			
統合ソフトウェア	Applixware	ネットワーク関連	
	ワープロ, 表計算, お絵描き, 電子メール		Netscape,gnus,mh-e
エディタ	mule	TeX	platex
数式処理	Mathematica	言語処理系	gcc,GNU pascal,squick
		画像処理	GIMP,xpaint
10206・32 情報処理教室 (10号館・3号館)			
Macintosh			
統合ソフトウェア	Claris Works	ネットワーク関連	
	ワープロ, 表計算, お絵描き, データベース, ドロー	Telnet	Better Telnet
キーボード練習	Typing Tutor	WWW	Netscape
		FTP	Fetch
21 情報処理教室 (2号館)			
Solaris			
エディタ	mule	TeX	latex
数式処理	Mathematica	言語処理系	gcc,GNU pascal
ネットワーク関連		画像処理	GIMP,xpaint
	Netscape,gnus,mh-e		
51 情報処理教室 (5号館)			
WindowsNT			
10号館 WindowsNT 環境と同じ		(追加分)	
		ワープロ	一太郎
		データベース	TheCARD
		経営シミュレーション	ステラ

### 3. 利用資格

本学の学生であれば自由に利用することができますが、利用機器によってはユーザ名とパスワードが必要となります。ユーザ名は入学時、全員（平成 11 年度以降入学者）に発行しています、詳しくは P. 308「計算機センター」を参照してください。

### 4. 利用時間

1.10 号館情報処理教室 平日 8:45~20:00 土曜日 8:45~17:00

但し、平日 18:00 以降、土曜日 12:00 以降は一部の指定教室のみ利用できます。

#### 2. サテライト教室

平日 8:45~18:00 土曜日 8:45~12:00

#### 3. 学部・学科の情報処理教室

所管の事務室にお問い合わせください。

情報処理教室の利用は授業を優先しますが、授業のない時間帯は研究や自習として自由に利用できます。10 号館情報処理教室では自習教室として指示されている教室で利用してください。サテライト教室及び学部・学科の情報処理教室では各建物の掲示板で「情報処理教室利用時間割表」及び補講掲示等で授業有無を確認のうえ、利用してください。なお、日曜日・祝日、夏季一斉休業期間および年末年始期間は使用できません。その他、保守等によるシステム停止日、清掃等による閉室日は、その都度掲示でお知らせします。

### 5. 入室方法

入室は、学生証を情報処理教室出入口のカード読み取り装置に通し開錠のうえ、入室してください。なお、C2・C3・C4、11 情報処理教室の利用時間は開錠しています。ただし、平日の午後 4 時 30 分以降および土曜日の正午以降に C2・C3・C4 の情報処理教室を利用する場合は、事前に計算機科学研究所事務室に届け出のうえ、鍵を借用してください。

### 6. 利用心得

利用者は、次に掲げる利用心得を厳守してください。

- (a) 教室内での飲食および喫煙の禁止。
- (b) 利用後は、機器の電源を切る。
- (c) 最終利用者はエアコンを止め室内を消灯する。
- (d) 教室内のマニュアルおよび備品の持ち出しの禁止。
- (e) ソフトウェアの複製および持ち出しの禁止。
- (f) ソフトウェアのインストールの禁止。
- (g) その他所管の学部等事務室から特に指示があった場合は、これに従うこと。

# 付録C章 著作権法 (抜粋)

タイプミスなどの誤りが含まれている可能性があることを予めご了承ください。

公布 昭和四五・五・六(法四八)

施行 昭和四六・一・一(附則)

改正 昭和五三法四九、昭和五六法四五、昭和五八法七八、昭和五九法二三・四六、昭和六〇法六二、昭和六一法六四・六五、昭和六三法八七、平成元法四三、平成三法六三、平成四法百六、平成五法八九、平成六法第一一二、平成七法第九一、平成八年法第一一七号、平成九年法八六号

## 第一章 総則

### 第一節 通則

(目的)

第一条 この法律は、著作物並びに実演、レコード、放送及び有線放送に関し著作者の権利及びこれに隣接する権利を定め、これらの文化的所産の公正な利用に留意しつつ、著作者等の権利の保護を図り、もつて文化の発展に寄与することを目的とする。

(定義)

第二条 この法律において、次の各号に掲げる用語の意義は、当該各号に定めるところによる。

一 著作物 思想又は感情を創作的に表現したものであつて、文芸、学術、美術又は音楽の範囲に属するものをいう。

二 著作者 著作物を創作する者をいう。

三 実演 著作物を、演劇的に演じ、舞い、演奏し、歌い、口演し、朗詠し、又はその他の方法により演ずること(これらに類する行為で、著作物を演じないが芸術的な性質を有するものを含む。)をいう。

四 実演家 俳優、舞踊家、演奏家、歌手その他実演を行なう者及び実演を指揮し、又は演出する者をいう。

五 レコード 蓄音機用音盤、録音テープその他の物に音を固定したもの(音をもつばら影像とともに再生することを目的とするものを除く。)をいう。

六 レコード製作者 レコードに固定されている音を最初に固定した者をいう。

七 商業用レコード 市販の目的をもつて製作されるレコードの複製物をいう。

七の二 公衆送信 公衆によって直接受信されることを目的として無線送信又は有線電気通信の送信(有線電気通信設備で、その一部の設置の場所が他の部分の設置の場所と同一の構内(その構内が二以上の者の占有に属している場合には、同一の者の占有に属する区域内)にあるものによる送信(プログラムの著作物の送信を除く。))を除く。)を行なうことをいう。

八 放送 公衆送信のうち、公衆によって同一の内容の送信が同時に受信されることを目的として行う無線通信の送信をいう。

九 放送事業者 放送を業として行なう者をいう。

九の二 有線放送 公衆送信のうち、公衆によつて同一の内容の送信が同時に受信されることを目的として行う有線電気通信の送信をいう。

九の三 有線放送事業者 有線放送を業として行なう者をいう。

九の四 自動公衆送信 公衆送信のうち、公衆からの求めに応じ自動的に行うもの(放送又は有線放送に該当する

ものを除く。)をいう。

九の五 送信可能化 次のいずれかに掲げる行為により自動公衆送信し得るようにすることをいう。

イ 公衆の用に供されている電気通信回線に接続している自動公衆送信装置（公衆の用に供する電気通信回線に接続することにより、その記録媒体のうち自動公衆送信の用に供する部分（以下この号において「公衆送信用記録媒体」という。以下同じ。）の公衆送信用記録媒体に情報を記録し、情報が記録された記録媒体を当該自動公衆送信装置の公衆送信用記録媒体として加え、若しくは情報が記録された記録媒体を当該自動公衆送信装置の公衆送信用記録媒体に変換し、又は当該自動公衆送信装置に情報を入力すること。

ロ その公衆送信用記録媒体に情報が記録され、又は当該自動公衆送信装置に情報が入力されている自動公衆送信装置について、公衆の用に供されている電気通信回線への接続（配線、自動公衆送信装置の指導、送受信プログラム等の起動その他の一連の行為により行われる場合には、当該一連の行為のうち最後のものをいう。）を行うこと。

十 映画製作者 映画の著作物の製作に発意と責任を有する者をいう。

十の二 プログラム 電子計算機を機能させて一の結果を得ることができるようにこれに対する指令を組み合わせたものとして表現したものをいう。

十の三 データベース 論文、数値、図形その他の情報の集合物であつて、それらの情報を電子計算機を用いて検索することができるように体系的に構成したものをいう。

十一 二次的著作物 著作物を翻訳し、編曲し、若しくは変形し、又は脚色し、映画化し、その他翻案することにより創作した著作物をいう。

十二 共同著作物 二人以上の者が共同して創作した著作物であつて、その各人の寄与を分離して個別的に利用することができないものをいう。

十三 録音 音を物に固定し、又はその固定物を増製することをいう。

十四 録画 映像を連続して物に固定し、又はその固定物を増製することをいう。

十五 複製 印刷、写真、複写、録音、録画その他の方法により有形的に再製することをいい、次に掲げるものについては、それぞれ次に掲げる行為を含むものとする。

イ 脚本その他これに類する演劇用の著作物 当該著作物の上演、放送又は有線放送を録音し、又は録画すること。

ロ 建築の著作物 建築に関する図面に従つて建築物を完成すること。

十六 上演 演奏（歌唱を含む。以下同じ。）以外の方法により著作物を演ずることをいう。

十七 口述 朗読その他の方法により著作物を口頭で伝達すること（実演に該当するものを除く。）をいう。

十八 上映 著作物を映写幕その他の物に映写することをいい、これに伴つて映画の著作物において固定されている音を再生することを含むものとする。

十九 頒布 有償であるか又は無償であるかを問わず、複製物を公衆に譲渡し、又は貸与することをいい、映画の著作物又は映画の著作物において複製されている著作物にあつては、これらの著作物を公衆に提示することを目的として当該映画の著作物の複製物を譲渡し、又は貸与することを含むものとする。

二十 国内 この法律の施行地をいう。

2 この法律にいう「美術の著作物」には、美術工芸品を含むものとする。

3 この法律にいう「映画の著作物」には、映画の効果に類似する視覚的又は視聴覚的效果を生じさせる方法で表現され、かつ、物に固定されている著作物を含むものとする。

4 この法律にいう「写真の著作物」には、写真の製作方法に類似する方法を用いて表現される著作物を含むものとする。

5 この法律にいう「公衆」には、特定かつ多数の者を含むものとする。

6 この法律にいう「法人」には、法人格を有しない社団又は財団で代表者又は管理人の定めがあるものを含むものとする。

7 この法律において、「上演」、「演奏」又は「口述」には、著作物の上演、演奏又は口述で録音され、又は録画されたものを再生すること（公衆送信又は上映に該当するものを除く。）を含み、「上演」、「演奏」、「口述」又は「上映」

には、著作物の上演、演奏、口述又は上映を電気通信設備を用いて伝達すること（公衆送信に該当するものを除く。）を含むものとする。

8 この法律にいう「貸与」には、いずれの名義又は方法をもつてするかを問わず、これと同様の使用の権原を取得させる行為を含むものとする。

9 この法律において、第一項第七号の二、第八号、第九号の二、第九号の四、第九号の五若しくは第十三号から第十九号まで又は前二項に掲げる用語については、それぞれこれらを動詞の語幹として用いる場合を含むものとする。

略

## 第二章 著作者の権利

### 第一節 著作物

（著作物の例示）

#### 第十条

この法律にいう著作物を例示すると、おおむね次のとおりである。

- 一 小説、脚本、論文、講演その他の言語の著作物
- 二 音楽の著作物
- 三 舞踊又は無言劇の著作物
- 四 絵画、版画、彫刻その他の美術の著作物
- 五 建築の著作物
- 六 地図又は学術的な性質を有する図面、図表、模型その他の図形の著作物
- 七 映画の著作物
- 八 写真の著作物
- 九 プログラムの著作物

2 事実の伝達にすぎない雑報及び時事の報道は、前項第一号に掲げる著作物に該当しない。

3 第一項第九号に掲げる著作物に対するこの法律による保護は、その著作物を作成するために用いるプログラム言語、規約及び解法に及ばない。この場合において、これらの用語の意義は、次の各号に定めるところによる。

- 一 プログラム言語 プログラムを表現する手段としての文字その他の記号及びその体系をいう。
- 二 規約 特定のプログラムにおける前号のプログラム言語の用法についての特別の約束をいう。
- 三 解法 プログラムにおける電子計算機に対する指令の組合せの方法をいう。

（二次的著作物）

第十一条 二次的著作物に対するこの法律による保護は、その原著物の著作者の権利に影響を及ぼさない。

（編集著作物）

第十二条 編集物（データベースに該当するものを除く。以下同じ。）でその素材の選択又は配列によって創作性を有するものは、著作物として保護する。

2 前項の規定は、同項の編集物の部分を構成する著作物の著作者の権利に影響を及ぼさない。

（データベースの著作物）

第十二条の二 データベースでその情報の選択又は体系的な構成によつて創作性を有するものは、著作物として保護する。

2 前項の規定は、同項のデータベースの部分を構成する著作物の著作者の権利に影響を及ぼさない。

（権利の目的とならない著作物）

第十三条 次の各号のいずれかに該当する著作物は、この章の規定による権利の目的となることができない。

- 一 憲法その他の法令
- 二 国又は地方公共団体の機関が発する告示、訓令、通達その他これらに類するもの
- 三 裁判所の判決、決定、命令及び審判並びに行政庁の裁決及び決定で裁判に準ずる手続きにより行なわれるもの
- 四 前三号に掲げるものの翻訳物及び編集物で、国又は地方公共団体の機関が作成するもの

略

### 第三節 権利の内容

#### 第一款 総則

( 著作者の権利 )

第十七条 著作者は、次条第一項、第十九条第一項及び第二十条第一項に規定する権利（以下「著作者人格権」という。）並びに第二十一条から第二十八条までに規定する権利（以下「著作権」という。）を享有する。

2 著作者人格権及び著作権の享有には、いかなる方式の履行をも要しない。

#### 第二款 著作者人格権

( 公表権 )

第十八条 著作者は、その著作物でまだ公表されていないもの（その同意を得ないで公表された著作物を含む。次項において同じ。）を公衆に提供し、又は提示する権利を有する。当該著作物を原著作物とする二次的著作物についても、同様とする。

2 著作者は、次の各号に掲げる場合には、当該各号に掲げる行為について同意したものと推定する。

一 その著作物でまだ公表されていないものの著作権を譲渡した場合 当該著作物をその著作権の行使により公衆に提供し、又は提示すること。

二 その美術の著作物又は写真の著作物でまだ公表されていないものの原作品を譲渡した場合 これらの著作物をその原作品による展示の方法で公衆に提示すること。

三 第二十九条の規定によりその映画の著作物の著作権が映画製作者に帰属した場合 当該著作物をその著作権の行使により公衆に提供し、又は提示すること。

( 氏名表示権 )

第十九条 著作者は、その著作物の原作品に、又はその著作物の公衆への提供若しくは提示に際し、その実名若しくは変名を著作者名として表示し、又は著作者名を表示しないこととする権利を有する。その著作物を原著作物とする二次的著作物の公衆への提供又は提示に際しての原著作物の著作者名の表示についても、同様とする。

2 著作物を利用する者は、その著作者の別段の意思表示がない限り、その著作物につきすでに著作者が表示しているところに従って著作者名を表示することができる。

3 著作者名の表示は、著作物の利用の目的及び態様に照らし著作者が創作者であることを主張する利益を害するおそれがないと認められるときは、公正な慣行に反しない限り、省略することができる。

( 同一性保持権 )

第二十条 著作者は、その著作物及びその題号の同一性を保持する権利を有し、その意に反してこれらの変更、切除その他の改変を受けないものとする。

2 前項の規定は、次の各号のいずれかに該当する改変については、適用しない。

一 第三十三条第一項（同条第四項において準用する場合を含む。）又は第三十四条第一項の規定により著作物を利用する場合における用字又は用語の変更その他の改変で、学校教育の目的上やむを得ないと認められるもの

二 建築物の増築、改築、修繕又は模様替えによる改変

三 特定の電子計算機においては利用し得ないプログラムの著作物を当該電子計算機において利用し得るようにするため、又はプログラムの著作物を電子計算機においてより効果的に利用し得るようにするために必要な改変

四 前三号に掲げるもののほか、著作物の性質並びにその利用の目的及び態様に照らしやむを得ないと認められる改変

### 第三款 著作権に含まれる権利の種類

#### (複製権)

第二十一条 著作者は、その著作物を複製する権利を専有する。

#### (上演権及び演奏権)

第二十二条 著作者は、その著作物を、公衆に直接見せ又は聞かせることを目的として(以下「公に」という。)上演し、又は演奏する権利を専有する。

#### (公衆送信権等)

第二十三条 著作者は、その著作物について、公衆送信(自動公衆送信の場合にあつては、送信可能化を含む。)を行う権利を専有する。

2 著作者は、公衆送信されるその著作物を受信装置を用いて公に伝達する権利を専有する。

#### (口述権)

第二十四条 著作者は、その言語の著作物を公に口述する権利を専有する。

#### (展示権)

第二十五条 著作者は、その美術の著作物又はまだ発行されていない写真の著作物をこれらの原作品により公に展示する権利を専有する。

#### (上映権及び頒布権)

第二十六条 著作者は、その映画の著作物を公に上映し、又はその複製物により頒布する権利を専有する。

2 著作者は、映画の著作物において複製されているその著作物を公に上映し、又は当該映画の著作物の複製物により頒布する権利を専有する。

#### (貸与権)

第二十六条の二 著作者は、その著作物(映画の著作物を除く。)をその複製物(映画の著作物において複製されている著作物にあつては、当該映画の著作物の複製物を除く。)の貸与により公衆に提供する権利を専有する。

#### (翻訳権、翻案権等)

第二十七条 著作者は、その著作物を翻訳し、編曲し、若しくは変形し、又は脚色し、映画化し、その他翻案する権利を専有する。

#### (二次的著作物の利用に関する原作者の権利)

第二十八条 二次的著作物の原著作物の著作者は、当該二次的著作物の利用に関し、この款に規定する権利で当該二次的著作物の著作者が有するもの同一の種類を専有する。

略

## 第五款 著作権の制限

### (私的使用のための複製)

第三十条 著作権の目的となつてゐる著作物(以下この款において単に「著作物」という。)は、個人的に又は家庭内その他これに準ずる限られた範囲内において使用すること(以下「私的使用」という。)を目的とする場合には、公衆の使用に供することを目的として設置されている自動複製機器(複製の機能を有し、これに関する装置の全部又は主要な部分が自動化されている機器をいう。)を用いて複製するときを除き、その使用する者が複製することができる。

2 私的使用を目的として、デジタル方式の録音又は録画の機能を有する機器(放送の業務のための特別の性能その他の私的使用に通常供されない特別の性能を有するもの及び録音機能付きの電話機その他の本来の機能に附属する機能として録音又は録画の機能を有するものを除く。)であつて政令で定めるものにより、当該機器によるデジタル方式の録音又は録画の用に供される記録媒体であつて政令で定めるものに録音又は録画を行う者は、相当な額の補償金を著作権者に支払わなければならない。

### (図書館等における複製)

第三十一条 図書、記録その他の資料を公衆の利用に供することを目的とする図書館その他の施設で政令で定めるもの(以下この条において「図書館等」という。)においては、次に掲げる場合には、その営利を目的としない事業として、図書館等の図書、記録その他の資料(以下この条において「図書館資料」という。)を用いて著作物を複製することができる。

- 一 図書館等の利用者の求めに応じ、その調査研究の用に供するために、公表された著作物の一部分(発行後相当期間を経過した定期刊行物に掲載された個々の著作物にあつては、その全部)の複製物を一人につき一部提供する場合
- 二 図書館資料の保存のため必要がある場合
- 三 他の図書館等の求めに応じ、絶版その他これに準ずる理由により一般に入手することが困難な図書館資料の複製物を提供する場合

### (引用)

第三十二条 公表された著作物は、引用して利用することができる。この場合において、その引用は、公正な慣行に合致するものであり、かつ、報道、批評、研究その他の引用の目的上正当な範囲内で行なわれるものでなければならない。

2 国又は地方公共団体の機関が一般に周知させることを目的として作成し、その著作の名義の下に公表する広報資料、調査統計資料、報告書その他これらに類する著作物は、説明の材料として新聞紙、雑誌その他の刊行物に転載することができる。ただし、これを禁止する旨の表示がある場合は、この限りでない。

## 略

### (プログラムの著作物の複製物の所有者による複製等)

第四十七条の二 プログラムの著作物の複製物の所有者は、自ら当該著作物を電子計算機において利用するために必要と認められる限度において、当該著作物の複製又は翻案(これにより創作した二次的著作物の複製を含む。)をすることができる。ただし、当該利用に係る複製物の使用につき、第一百三十二条第二項の規定が適用される場合は、この限りでない。

2 前項の複製物の所有者が当該複製物(同項の規定により作成された複製物を含む。)のいずれかについて滅失以外の事由により所有権を有しなくなつた後には、その者は、当該著作権者の別段の意思表示がない限り、その他の複製物を保存してはならない。

### (出所の明示)

第四十八条 次の各号に掲げる場合には、当該各号に規定する著作物の出所を、その複製又は利用の態様に応じ合理

的と認められる方法及び程度により、明示しなければならない。

略

#### 第四節 保護期間

(保護期間の原則)

第五十一条 著作権の存続期間は、著作物の創作の時に始まる。

2 著作権は、この節に別段の定めがある場合を除き、著作者の死後(共同著作物にあつては、最終に死亡した著作者の死後。次条第一項において同じ。)五十年を経過するまでの間、存続する。

略

(著作物の利用の許諾)

第六十三条 著作権者は、他人に対し、その著作物の利用を許諾することができる。

2 前項の許諾を得た者は、その許諾に係る利用方法及び条件の範囲内において、その許諾に係る著作物を利用することができる。

3 第一項の許諾に係る著作物を利用する権利は、著作権者の承諾を得ない限り、譲渡することができない。

4 著作物の放送又は有線放送についての第一項の許諾は、契約に別段の定めがない限り、当該著作物の録音又は録画の許諾を含まないものとする。

5 著作物の送信可能化について第一項の許諾を得た者が、その許諾に係る利用方法及び条件(送信可能化の回数又は送信可能化に用いる自動公衆送信装置に係るものを除く。)の範囲内において反復して又は他の自動公衆送信装置を用いて行う当該著作物の送信可能化については、第二十三条第一項の規定は適用しない。

略

(侵害とみなす行為)

第一百三条 次に掲げる行為は、当該著作人的人格権、著作権、出版権又は著作隣接権を侵害する行為とみなす。

一 国内において頒布する目的をもつて、輸入の時に国内で作成したとしたならば著作人的人格権、著作権、出版権又は著作隣接権の侵害となるべき行為によつて作成された物を輸入する行為

二 著作人的人格権、著作権、出版権又は著作隣接権を侵害する行為によつて作成された物(前号の輸入に係る物を含む。)を情を知つて頒布し、又は頒布の目的をもつて所持する行為

2 プログラムの著作物の著作権を侵害する行為によつて作成された複製物(当該複製物の所有者によつて第四十七条の二第一項の規定により作成された複製物並びに前項第一号の輸入に係るプログラムの著作物の複製物及び当該複製物の所有者によつて同条第一項の規定により作成された複製物を含む。)を業務上電子計算機において使用する行為は、これらの複製物を使用する権原を取得した時に情を知つていた場合に限り、当該著作権を侵害する行為とみなす。

3 著作者の名誉又は声望を害する方法によりその著作物を利用する行為は、その著作人的人格権を侵害する行為とみなす。

以下略

## 付録D章 参考文献

ここにあげる文献にある記述が全て cc 環境で適用できるとは限らないことに注意してください。

—— Unix 全般、シェル、コマンドなどについて ——

たのしい UNIX -UNIX への招待- 坂本 文著：アスキー出版局刊  
月刊雑誌 UNIX Magazine の連載を集成した UNIX 初心者向けの入門書。

続 たのしい UNIX -シェルへの招待- 坂本 文著：アスキー出版局刊  
上の一冊の続編。今度はシェルについて解説してくれる。

実用 UNIX ハンドブック 舟本 奨著：ナツメ社  
UNIX コマンドの簡単なリファレンス。

UNIX & X コマンド辞典 Alan Southerton, Edwin C. Perkins, Jr. 著、加藤大典訳：丸善、1995年7月  
刊、3,800円

UNIX コマンドの非常に詳細なリファレンス。比較的高価ではあるが、400ページ以上にわたり各種のコマンドに関して、およそありとあらゆる例が挙げてある。多くのユーザのコマンドの使い方を試す手間をかなり省いてくれるだろう。ざーっと眺めて新しいコマンドの使い道を発見するのもよし。

DOS ユーザのための UNIX 入門 Douglas W. Topham 著、中西隆訳：技術評論社、1991年刊、1900円  
タイトル通り、DOS ユーザの視点から見た UNIX の機能説明です。後半はむしろシステム管理社向けに書いてあります。

UNIX C SHELL フィールドガイド G・アンダーソン、P・アンダーソン著：落水 浩一郎、大木 敦  
雄訳：パーソナルメディア刊  
csh のほぼ完全なガイド。

UNIX step++ シェルプログラミングのコツ 西沼 行博著：マグロウヒル刊  
残念ながら csh ではなく、sh についての説明が主体。記述も少々古いがシェルを使いこなしたい人には便利。

The Unix Super Text 山口 和紀監修、于 旭・中村 敦司・新城 靖・西山 博泰・古瀬 一隆・石川 佳  
治・佐々木 重雄・林 謙一・荻原 一隆・金谷 英信・鈴木 孝幸・黒石 和宏 著：技術評論社刊  
BSD と呼ばれるタイプの UNIX 主体に書いてあるため、cc 環境とは相違がある場合が散見されるが、一般的な概念からハウツーまで、広い分野に関して教えてくれる。上下巻の二冊組。上巻は一般的な UNIX の使い方、電子メール、ネットニュース、X ウィンドウなどについて。下巻は L<sup>A</sup>T<sub>E</sub>X、プログラミング、システム管理などについて。高価なのが難点か。

誰にでも使える UNIX 講座 安岡孝一著：ソフトバンク社刊、1992年  
yasuoka さんが root さんに色々教えてもらう対話形式の本。login, logout から始まって UNIX

の基本的なツール sh, csh, sed, make, awk の使い方がこれ一冊でわかる。UNIX を使いこなしてみたい人の入門書。

MH & xmh Jerry Peek 著、倉骨彰 訳、砂原秀樹・鈴木麗 監訳：アスキー出版局刊、1994年11月、5,800円

MH に関するマニュアル。

UNIX の環境設定 久野禎子、久野靖著：アスキー出版、1993年9月刊、1,800円

cc 環境は最初の段階で殆ど全ての環境設定が行われていますが、この本は様々な UNIX での機能設定の方法について説明されています。これから自分の環境を構築したい、しなければならない人向けといえるでしょう。

#### —— Mule について ——

入門 Mule 大木 敦雄著：アスキー出版刊

まさに Mule の入門書。

Mule でじゃん! だって UNIX ですよ 平山弘之著：メロン出版刊、1,600円

おじさんには頭がいたくなるような題名ですが、内容もそのとおり、くだけたもので、計算機を専門としない超ビギナーには、面白く読めるものと思います。mule に関する一切の難しいことが、ばっさり省略されている ところがこの本の特徴です。

#### —— Emacs について ——

Emacs は Mule の前のバージョンです。基本的な操作などに付いては殆どこの Emacs の文献が利用できます。参考にしてください。

GNU Emacs 入門 (株) 京都ソフトウェアリサーチメディアデザイン編：オーム社刊、1994

Emacs を非常に分かり安く解説しています。自習にも向いていると思います。付録としてついている切り取り式のコマンド一覧も便利。

GNU Emacs Debra Cameron and Bill Rosenblatt 著：ハイパーウェア監訳：ソフトバンク株式会社刊

Emacs のほぼ完全なガイド。NutsShell (ナッツ(どんぐり)のカラ?) シリーズと呼ばれる非常に詳細な Unix 関係のドキュメントのシリーズの一冊。

GNU Emacs 完全ガイド M.A. シュノーバー、J.S. ボウイ、W.R. アーノルド著・小畑喜一、磯谷正孝、山野修、林秀幸訳：アジソン・ウェスレイ・トッパン刊 (情報科学シリーズ 30,35) (上巻：1992年11月 3,400円、下巻：1993年3月 3,900円)

本書が取り上げている GNU Emacs は Version が、18.57 です。紐解く時に「完全ガイド」の安心感があります。

GNU Emacs マニュアル Richard Stallman 著：竹内郁雄・天海良治訳：共立出版刊、1988年2月、2,930円

GNU の御大自らの解説です。《古典》の部類なのかもしれませんが。

入門 NEmacs 大木 敦雄著：アスキー出版刊

Emacs だけでなく、EGG, MHE, GNUS などについても説明してくれている。

—— TeX について ——

楽々 $\LaTeX$  野寺 隆志著：共立出版刊、第二版、1994、2,900 円

$\LaTeX$  を用いた TeX の入門書。初心者には最適。

$\LaTeX$  入門-美文書作成のポイント- 奥村 晴彦監修：技術評論社刊、1994

TeX についていろいろ丁寧に教えてくれる。

やさしい $\LaTeX$  のはじめかた すずき ひろのぶ著：オーム社刊、1991

表題通り $\LaTeX$  のやさしい入門書です。その割に、便利なのがチョコチョコと書いてある。

日本語 $\LaTeX$  定番スタイル集 No.1, No.2 鷺谷 好輝著：インプレス刊

京都産業大学でキャンパスライセンスを取得している $\LaTeX$  のスタイルファイルの使い方解説書。きれいなスタイルファイルの見本としてもよい。

$\LaTeX$  エラーマニュアル 今井豊著：カットシステム (Tel.0423-94-2218) 刊、1994 年 6 月、2,300 円

マイナーなところからの出版のためか、話題になりませんでした。「エラーをなおし、エラーから学ぶ本格的解説書」ではあります。AUCTeX があれば、半減しているのでしょうか、TeX を使う楽しさの半分は、「エラーとの格闘」ゲームではないでしょうか。その意味では、ゲーム本に分類すべきなのかもしれませんが。

てくてく TeX 阿瀬はる美著：アスキー出版局刊、(上巻 1994 年 11 月、2,000 円、下巻 1994 年 12 月、2,000 円)

語り口の楽しさが、坂本文著「たのしい UNIX」に似ているのは、「UNIX MAGAZINE」1989/04-90/07 連載だったせいでしょうね。

逆引き $\LaTeX$  D.J. パーガー著、引地 信之・引地 美恵子訳：マグロウヒル刊

「こんなことができるのではないか？」と思った機能から、その方法を調べる。

$\LaTeX$  トータルガイド 伊藤 和人著：秀和システムトレーディング刊

$\LaTeX$  技能の華麗な一覧表。

文書処理システム $\LaTeX$  Leslie Lamport 著、Edgar Cooke・倉沢 良一監訳、大野 俊治・小暮 博道・藤浦 はる美著：アスキー出版局刊

他の $\LaTeX$  解説書の説明では納得できない時の駆け込み寺。入門書なのに理論的。

改訂新版 TeX ブック Donald E. Knuth 著、斉藤信男監修、鷺谷好輝訳：アスキー出版局刊

教祖様直筆 TeXnician の Bible。頭から噛じるに困難でも、昼寝の枕に最適。

日本語 $\LaTeX 2\epsilon$  ブック 中野 賢 著：株式会社アスキー刊

株式会社アスキーが日本語版で縦組ができる $\LaTeX$  をリリースし、現在 p $\LaTeX 2\epsilon$  という名前で公開され、それに対するマニュアル本と位置づけられるでしょう。

$\LaTeX 2\epsilon$  階梯 藤田 眞作 著：アジソン・ウェスレイ・パブリッシャーズ・ジャパン刊

$\LaTeX 2\epsilon$  にかなり詳しい参考書。 $\LaTeX 2\epsilon$  と本気で取り組むのなら是非揃えておくべき。

—— インターネットについて ——

インターネット 村井純著：岩波書店、1995 年 11 月刊、650 円

岩波の新書版です。WIDE 代表、日本のミスターインターネットとも言える村井氏がインターネッ

トの経緯、理念などについて述べています。多分に技術的な内容を含んでおり、技術と理念が同時に語られているところが良いと思います。

ハッピー・ネットワークング 山本和彦著：アスキー出版局、1994年7月刊、1,500円

NEmacs, mail, news, FTP だけに焦点を絞った若者による新入生向けであることに好感が持てました。cc 環境では殆ど等価のドキュメントを ccinfo コマンドで印刷させることもできますが、綺麗に製本された物が欲しければやはりこれを買うことになるでしょう。

インターネット情報生活入門 グループまたたび著：技術評論社、1994年10月刊、1,700円。 インターネットの全体的な紹介をビジュアルにしている点がよいと思います。

インターネット参加の手引き～1994年度版 WIDE Project 編・村井純、吉村伸 監修：共立出版、bit 別冊、1994年7月刊、4,800円

情報はタツプリ詰まっていますけど、高価。1995年度版が出るそうです。

インターネットユーザーズガイド Ed Krol 著・村井純 監訳：インターナショナル・トムソン・パブリッシング・ジャパン刊 1994年5月刊、4,400円

ブームに向けてタイミング良く出版されてスタンダードなものになったようですけれど悪評を批判されたりもしていました。1995年の1月に「改訂版」が出ていましたが、訳の改善がはたしてどの程度行なわれたのか、確認していません。(これでは紹介文にならない!)

インターネット・ナビゲータ Paul Gilster 著・菱山博陸訳：丸善、1995年2月刊、6,592円

原書第2版が底本。「本書は常にモデムでインターネット利用する人を念頭においています。」(p.vii)って台詞に、コロっと心底、騙されました。インターネット論としても優れているものだと思います。

#### —— WWW ページ作成、HTML について ——

インターネット ホームページデザイン 吉村信、家永百合子、鏡聡：翔泳社、1995年6月刊、2,400円

HTML 文法のリファレンス。かなり細かく書いてあり、殆ど全ての HTML 文法をカバーしていると思われる。各文法がどのクライアント向けの拡張なのか、即ちどのクライアントソフトでならどのような効果が出るのかについて特に注意が払われているので、多くの人にページを見て欲しいページ制作には欠かせません。

HTML 入門 WWW ページの作成と公開 ローラ・リメイ著、武舎広幸、久野禎子、久野靖訳：プレントニスホール出版、1995年12月刊、3,900円

WWW ページ制作に関するノウハウと技術がぎっしり詰まっています。ページのデザインに関しての助言も数多く、バランスの取れたページデザインや構成をする上で一度は見るのがお勧めです。

続 HTML 入門 新機能、CGI、Web の進化 ローラ・リメイ著：武舎広幸、久野禎子、久野靖訳：プレントニスホール出版、1995年12月刊、3,900円

上記「HTML 入門」の続編です。前作の出版以降に追加された機能について解説しています。特に CGI (Common Gateway Interface) を使って、プログラムによる機能を WWW に組み込む方法の紹介が含まれています。

# 付録E章 トラブル解決

情報処理教室のコンピュータを使っている時に起こる可能性のあるトラブルと、その対処について説明します。コンピュータがおかしくなり、自分の手に負えないと思ったときにコンピュータの電源を切ったりしていませんか？それはコンピュータの故障の原因になりますので止めましょう。なりより、あなたが正しいコンピュータの知識を得るチャンスなのにそれを放棄していることにもなります。トラブルの解決方法を知っていて損はありませんよ。

## E.1 画面が正しく表示されない

キーボードを確認する

アルファベットを入力すると文字が大文字で出てくる、日本語を入力するとカタカナになる等、ちょっとだけおかしい時はキーボードの特殊なキーが押されているときです。CAPS LOCK キーやひらがなキー、Num Lock キーを押してみると状況が変わるでしょう。なお、WindowsNT コンピュータのキーボードでは CAPS LOCK キーはシフトキーを押しながら押します。

Mule の利用中

単に画面が乱れているだけなら、C-1 (エル) をしてみましょう。画面を再描画してくれるので正常になるはずです。文字を入力すると全然違う文字になってしまうときは Mule がおかしくなっています。<ESC> を一度押した状態で日本語を入力したときなどによく起こる現象です。mule を一度終了するのが確実ですが、<ESC> を押して「あ<sup>1</sup>」を入力すると直ることがあります。

Prompt 表示中 (コマンド入力画面)

cat コマンドで変なデータを含むファイルを表示してしまった時など、何を入力しても変な漢字だらけの画面になってしまうときは clear コマンドを試してください。それでもだめなら cls<sup>2</sup> コマンドで直るでしょう。それでも駄目なら logout して (画面がおかしいだけで logout コマンドは受け付けるはず) login し直すと大丈夫でしょう。

## E.2 フロッピーディスクが取り出せない

MacOS 編

Macintosh の正しいフロッピーディスクの取り出しかたは、デスクトップ上のフロッピーディスクのアイコンをドラッグしてごみ箱に入れることです。これで出てこないならフロッピーに保存された書類から

---

<sup>1</sup> なんでもいいですが日本語一文字を入力します。

<sup>2</sup> cc 環境のコマンドです。一般的なものではありません。

アプリケーションを立ちあげていませんか？作業中の書類を保存して、一旦アプリケーションを終了させてください。取り出せるようになります。

コンピュータが止まってしまったように見えるならこの次のフリーズ、MacOS 編を参照してください。フリーズを解決するとフロッピーディスクも取り出せるようになります。

## UNIX 編

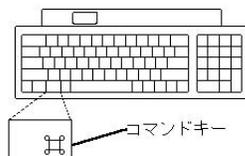
一度 logout してください。logout 時にフロッピーディスクが出てきます。出てこないときはもう一度 login して取り出し作業をすれば大丈夫でしょう。

## E.3 コンピュータが止まった

いつもどおり使っていて、急にコンピュータが止まってしまい、キーボードもマウスも使えなくなる時があります。この状態をフリーズ、またはハングアップといいます。

### MacOS 編

よくある原因に、キーボードの配線がはずれている（はずれかかっている）ということがあります。まず最初にこれを調べてください。問題がなければ以下を順番に試して行ってください。



なにかアプリケーションを使っている途中でフリーズしたのであれば、まず「そのアプリケーションだけを強制終了する」ことを試みます。なお、当該アプリケーションで作成していた文章は失われることとなります。command キーと option キーを押しながら、esc キーを押してください。うまく行くと、当該アプリケーションを終了してよいか確認する画面が出てきますので終了させてください。その後作業が続けられる状態に戻ると思いますが、なおシステム自体がダメージを受け不安定になっている可能性がありますので、念のため再起動させてから作業を続けるのが安全でしょう。

さて、上記キー組み合わせを試しても、Macintosh に変化がない場合、また「Finder を終了しますか？」などと表示されてしまった場合は、次に再起動を試みることになります。なお再起動により、作成中の書類は全て失われることとなります。command キーと control キーを押しながら Power キー（電源を入れる時に押すキー）を押して下さい。うまく行くと、一度画面が真っ暗になると同時に起動音が聞こえてくるでしょう。起動後は通常どおりに利用できます。

このキー組み合わせを試しても、まだ Macintosh になにも変化がない場合、また再起動はかかったものの、画面中央に「？」マークがついたフロッピーアイコンが表示されたまま動作しない場合はかなり重傷です。もう強制的に電源を切断するしかありません。以下に方法を記述しますが、あくまでも最終手段ですので（最悪の場合システムが破損します）普段は絶対に行わないでください。

- 11 情報処理教室の場合（Power Macintosh6100）

Macintosh6100 本体のフロッピーディスク差込口右下に電源ボタンがあります（触らないで等のシールがはってあります）。これをぐっと押し込んで下さい。本体の電源が切れます。その後、もう一度同じボタンをぐっと押ししてください。起動音が聞こえたら成功です。起動後は通常どおりに利用できます。

- 32 情報処理教室の場合（Power Macintosh8500）

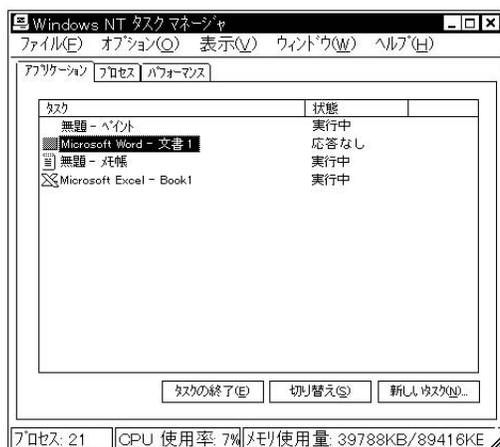
Macintosh8500 本体前面右下（グリーンのLEDランプ直下）に電源ボタンがあります（触らないで等のシールがはってあります）。これをぐっと押ししてください。本体の電源が切れます。その後、もう一度同じボタンをぐっと押ししてください。起動音が聞こえたら成功です。起動後は通常どおりに利用できます

## WindowsNT 編

Ctrl キー、Alt キー、Delete キーを同時に押すとウインドウが開きます<sup>3</sup>。



そこからタスクマネージャのボタンを押します。現れたウインドウからアプリケーションのタグを選び、コンピュータが止まった原因になったと思われるプログラムを選択してタスクの終了ボタンを押します。これで復活できるでしょう。復活したら一度 logout してコンピュータの電源を切ることをお勧めします。



## UNIX 編

UNIX の場合、フリーズしたように見えても実は動いている時が多いのです。動いているのなら、止まったコンピュータの目の前のキーボードやマウスは動かなくても、違うコンピュータからネットワークで止

<sup>3</sup> 開かないときはよほどひどく壊れている状態です。コンピュータが壊れないことを祈りながら電源を切るしかありません。

まった（ようにみえる）コンピュータに乗り込んでフリーズさせている原因のプログラムを止めることで解決できます。

まず近くのコンピュータに login します。説明を具体的にするためにここでは止まった（ように見える）コンピュータを cssol001、近くのコンピュータを cssol002 とします。また、cssol001 を使っていたユーザを ozaki とします。

cssol002 に login したら、`rlogin cssol001` として cssol001 にリモートログインします。

```
cssol002(81)% rlogin cssol001
Last login: Sat Mar 14 10:14:42 from cssol023
Sun Microsystems Inc. SunOS 5.5.1 Generic May 1996
cssol001(81)%
```

プロンプトが cssol001 のものになって、リモートログインに成功したことが判ります<sup>4</sup>。これでもう cssol002 のキーボードから cssol001 を使っているのです。続いて `ps` コマンドで、今動いているプロセス（プログラム）を表示させ、かつパイプを用いて `grep` コマンドで ozaki を含む行だけを抜き出します。

それ以外の機種は下記の表に対応するオプションを指定してください。

```
cssol001(81)% ps -auxw | grep ozaki5
ozaki  24908  0.2  0.1  972  836 pts/12  0 14:06:57 0:00 ps -auxw
ozaki  24909  0.1  0.1  708  484 pts/12  S 14:06:57 0:00 grep ozaki
ozaki  29506  0.1  0.2 1080  996 pts/12  S 09:11:17 0:01 -tcsh
ozaki  25830  0.0  0.2 1084  948 pts/10  S 08:30:45 0:00 -tcsh
ozaki  25931  0.0  0.2 1080  940 pts/13  S 08:31:09 0:00 mule
ozaki  26171  0.0  0.5 4472 3280 pts/13  S 08:31:19 0:41 mule
ozaki  26196  0.0  0.2 1080  952 pts/14  S 08:31:52 0:01 -tcsh
```

ユーザ ID の次に来る数字がプロセスの通し番号（プロセス ID）です。止まった原因になったと思われるプロセスにあたりをつけて<sup>6</sup>、そのプロセス ID を指定した `kill` コマンドを実行します。ここでは 8:31:19 に起動した mule が怪しいとしました。

```
cssol001(82)% kill 26171
```

これで、もし止まらない場合はもう少し強力にして止めるオプションをつけます。

```
cssol001(83)% kill -9 26171
```

これで実際の cssol001 の画面を見て、正常に戻れば大当たりです。変化がないようなら別のプロセスを止めてみましょう。

---

<sup>4</sup> いつまで経っても目的のコンピュータにつながらないときは計算機センターに連絡してください。このような状態でも電源を切つてはいけません。

<sup>5</sup> うまくいかないときは `-aux` の部分を `-ef` に変えてください。また、パイプ (`|`) は縦棒の記号です。いちやエルではありません。ほとんどのキーボードではシフトを押しながら `¥` キーで出るでしょう。

<sup>6</sup> 経験で判ってきます。それまでは順番に選びましょう

# 索引

.aux, 264  
.dvi, 264  
.log, 264  
.tex, 264  
<!--と-->, 212  
<A HREF= >, 222  
<A NAME= >, 223  
<ADDRESS>, 211  
<B>, 214  
BGCOLOR, 225  
<BIG></BIG>, 227  
<BLOCKQUOTE>, 219  
<BODY>, 210  
<BODY>, 225  
<BR>, 212  
<CAPTION>, 229  
<CENTER>, 228  
<DD>, 216  
<DIV ALIGN>, 228  
<DIV ALIGN=right>, 228  
<DL>, 216  
<DT>, 216  
<FONT SIZE=n>, 227  
<FONT COLOR=#?????>, 226  
<HEAD>, 210  
<HR>, 213  
<HTML>, 210  
<Hn>, 214  
<I>, 215  
<IMG SRC= ALT= >, 221  
<LI>, 215, 216  
<OL>, 216  
<P>, 212  
<PRE>, 219  
<SMALL></SMALL>, 227  
<SUB></SUB>, 227  
<SUP></SUP>, 227  
<TABLE>, 228  
<TD>, 229  
<TH>, 229  
<TITLE>, 210  
<TR>, 228  
<TT>, 215  
<UL>, 215  
¥author, 262  
¥bf, 261  
¥caption, 273  
¥date, 262  
¥em, 261  
¥fbox, 265  
¥footnote, 263  
¥footnotesize, 260  
¥framebox, 265  
¥hspace, 253  
¥indext, 253  
¥input, 289  
¥it, 261  
¥label, 263  
¥large, 260  
¥maketitle, 262  
¥marginpar, 263  
¥indent, 253  
¥normalsize, 260  
¥rm, 261  
¥sc, 261  
¥sf, 261  
¥sl, 261  
¥tiny, 260  
¥title, 262  
¥tt, 261  
¥underbar, 275  
¥verbatim, 252  
¥vspace, 254  
10 情報処理教室, 62  
11 情報処理教室, 42  
21 情報処理教室, 34

31 情報処理教室, 42  
32 情報処理教室, 42

ALINE, 226  
ALT, 221  
anonymous FTP, 155  
archie, 157  
AUCTeX, 290

bc, 75  
BetterTelnet, 42

C-, 23, 72  
Can't open display, 380  
cat, 84  
Cc:, 122  
cc 環境, 15  
center, 260  
chmod, 179  
command 索引, 358  
cp, 86, 105

date, 73  
description, 256  
Dired モード, 202  
DOCTYPE 宣言, 210  
dvi2ps, 244

EGG, 195  
enumerate, 256  
env, 183  
epsbox.sty, 272  
EUC コード, 149

FAQ, 371, 372  
fetch, 150  
figure 環境, 272  
finger, 74  
fj, 379  
flushleft, 260  
flushright, 260  
Followup-To, 380  
FreePPP, 328  
FTP, 153  
FTP サイト, 224  
ftp, 167

GIF, 220  
GNUS の起動, 132  
grep, 373

hostname, 74  
HTML, 205  
html-helper-mode, 230  
HTML4.0, 233  
HTML 文法チェック, 231

index-j.html, 207, 208  
itemize 環境, 256

jarticle, 247  
jbook, 248  
JIS コード, 149  
jlatex, 242  
jreport, 247

kill, 373  
kyoto-su.ac.jp, 115

label, 263  
LaTeX, 241  
LINK, 374  
login, 22  
logout, 22, 40, 69  
lpr, 95, 245  
ls, 83

M-, 72  
Macintosh, 42  
Macintosh の手引き, 43, 47  
MacPPP, 328  
mail アドレス, 114  
mail サービス, 112  
man, 76  
master file, 294  
math.h, 382  
minipage 環境, 268  
みんなで使うコンピュータ, 15  
more, 85  
Mule, 87, 190  
Mule のエラーメッセージ, 88  
mule の環境設定, 376  
Mule の終了, 92

mv, 85, 107

Netscape, 207

news サービス, 113

nkf, 149

nup, 287

Permission denied, 177

PPP, 306

ps, 373

pwd, 102

quota, 172

quotation 環境, 255

quote 環境, 255

rm, 86, 107

root menu, 37

save, 92

script, 84, 373

setenv, 183

SGML, 235

signature, 128

SJIS コード, 149

Subject:, 122

Suspended, 79

tabular 環境, 270

telnet, 31

tnn, 379

To:, 122

tree, 101

Turbo Linux, 62

Ultra1, 34

Undo, 194

UNIX, 15

UNIX コマンド, 358

UNIX それから, 71

UNIX への道, 33

UNIX もっともっと, 169

URI, 159, 206

URL, 159

URL の検索, 159

verbatim 環境, 252

whois, 173

WindowsNT, 15

Wnn, 194

World Wide Web, 158

WS FTP, 152

WWW, 110

wwwmkdir, 206

WWW サービス, 113

xdvi, 243

XML, 235

X ウィンドウ, 31

X 環境で Mule, 203

yppasswd, 38, 45, 57, 67

ア－ティクル, 132

アウトラインマイナーモード, 295

アクセス権, 177

アクセスサーバ, 306

アクセント, 275

圧縮, 323

アンカー, 222

一連のコマンド, 188

イベント, 82

印刷, 95

印刷が薄い, 97

インターネット, 111

引用, 124, 255

引用のマナー, 145

引用符が付けられない, 377

インライン画像, 220

ウイルス, 32

ウィンドウ, 25, 190

ウィンドウ環境, 24

絵, 272

エコーライン, 190

エラーの種類, 285

エラーの対処, 282

遠隔操作, 31

円記号, 247

オプション, 74

カーソル, 72

カーソル移動, 193

カーソルの移動, 90

カードロックシステム, 50  
改行する, 212  
改行, 253  
改行コード, 148  
解凍, 323  
改ページ, 253  
拡張子, 208  
箇条書き, 256  
カスタマイズ, 203  
下線, 275  
画像を入れよう, 220  
(ことえりで)カタカナしか出ない, 382  
カット, 193  
紙詰まり, 97  
画面が乱れた, 193  
画面の移動, 91  
画面分割, 201  
カレントディレクトリ, 102  
環境変数, 183  
漢字 Talk7.5, 328  
関数, 281  
キー表記, 71  
キーボード, 22  
記号, 275  
記号入力モード, 198  
記事, 132  
記事を全部読んだことにする, 143  
脚注, 263  
キャンセル, 140  
キャンセルについて, 145  
行の移動, 91  
ギリシャ文字, 280  
空白, 252, 275  
区切り線を入れる, 213  
矩形, 193  
クロスポスト, 379  
計算機センターメインカウンター, 13  
検索, 201  
語句の説明, 72  
コマンド, 73  
コマンド索引, 358  
コマンド名を調べる, 77  
コメント, 212  
コンパイル, 243  
サブコマンド, 76  
参照, 263  
シェル, 80, 181  
シグネチャ, 376  
字下げ(インデント), 219  
実行結果をプリントアウト, 373  
シャットダウン, 40, 69  
ジャム, 97  
章, 248  
小小節, 248  
小節, 248  
小段落, 248  
情報検索, 173  
書体, 261  
シンボリックリンク, 374  
情報処理教室, 50  
垂直方向の空白, 254  
水平方向の空白, 253  
数式記号, 277  
数式モードの支援, 296  
スタイルファイル, 248  
スタイルファイル, 289  
セーブ, 92  
正規表現, 373  
セクション, 78  
節, 248  
セル, 229  
相互参照, 263  
メインカウンター, 13  
そのまま表示, 219  
そのまま出力, 252  
そのまま出力する, 252  
ターミナル, 31  
ダイヤルアップ IP 接続 (Mac), 328  
タイトル, 262  
ダイヤルアップ IP 接続, 304  
ダイヤルアップ IP 接続 (Win95), 308  
ダイヤルアップ IP 接続 (Win98), 315  
ダイヤルアップ IP 接続トラブル対策, 341  
タグ, 209  
単語登録, 200  
端末, 31  
段落を区切る, 212  
段落, 248

置換, 201  
チュートリアル, 93, 191  
中央寄せ, 260  
次の候補, 197  
次の文節, 196  
ツリー, 101  
ディスクの利用量, 172  
ディレクトリ, 99  
テキスト形式, 147  
電子メール, 110  
投稿, 132  
投稿する, 136  
ドキュメントスタイル, 247  
特殊文字 (HTML), 211  
特殊な文字, 197  
特殊文字 ( $\Gamma E30FLaTeX$ ), 250  
トップディレクトリ, 101  
トラブル, 28, 372  
トラブル解決, 398  
トラブルからの脱出, 78  
長さの単位, 255  
日本語入力, 194  
ニュースグループ, 132, 379  
ニュースグループのソート, 377  
ニュースグループを選ぶ, 135  
ネットニュース, 110  
ネットニュースの読み書き, 132  
ネットワーク, 111  
ネットワークサービス, 109  
ネットワークの世界へようこそ, 109  
パーセント記号, 252  
ハードリンク, 374  
パーミッション, 179, 381  
背景の色, 225  
バイナリ形式, 147  
ハイパーテキスト, 205  
パイプ, 187  
箱, 265  
パス, 102  
パスワード, 15, 19  
パスワードを変える, 38, 45, 57, 67  
パソコンから UNIX, 30  
バックスラッシュ, 247  
バッファ, 190  
バッファリスト, 201  
引数, 74  
ヒストリ, 81  
左寄せ, 260  
否定型, 278  
表, 228, 270  
表題, 262  
部, 248  
ファイル名の付け方, 208  
ファイル, 83  
身に覚えのないファイル, 171  
ファイルの一覧を見る, 83  
ファイルの移動, 107  
ファイルの印刷, 95  
ファイルの大きさ, 169  
ファイルのコピー, 105  
ファイルの消去, 86, 107  
ファイルの整理, 169  
ファイルの内容を見る, 84  
ファイルの名前を変える, 107  
ファイルの複写, 86  
ファイルの呼びだし, 192  
ファイルへの保存, 92  
ファイル名の指定, 90  
ファイル名の変更, 85  
ファイルを編集, 87  
ファイルを分けて編集, 289  
フィルタコマンド, 186  
フェンスモード, 195  
フォーマット, 146  
フォロー, 139, 145  
複数行のコメント, 296  
部分印刷, 286  
ブラウザ, 160  
プリンタ, 95, 349, 381  
プリンター一覧, 349  
プリンタのトラブル, 97  
プリントアウト, 244  
古い記事を読み返す, 141  
プロセス, 79  
プロセスの終了, 373  
(Mac で) フロッピーが取り出せない, 382  
フロッピーディスクの種類, 146  
フロッピーディスクの利用, 146

プロンプト, 73  
文節の区切り, 197  
ペースト, 193  
編集, 90  
（Mule で）変になったら, 192  
ホームディレクトリ, 101  
ホームページ, 158  
傍注, 263  
ポスト, 132  
ホストコンピュータ, 31  
ポストスクリプト, 244  
保存, 92, 194  
（フロッピーに）保存できない, 382  
ボタン, 28  
マウス, 24  
マクロ, 288  
マック, 42  
マックのリセットスイッチ, 382  
真ん中寄せ, 228  
右寄せ, 228, 260  
見出しの種類, 248  
メール, 110  
メールアドレス, 114  
メールでの返事の作法, 124  
メールの印刷, 127  
メールの実体, 126  
メールの整理, 125  
メールの返事を書く, 123  
メールを書く, 122  
メールを書くときの注意, 129  
メールを破棄, 123  
メールを日付順にする, 377  
メールを振り分ける, 376  
メールを読み返す, 122  
メールを読む, 119  
モードライン, 190, 202  
文字コード, 148, 375  
文字コードの変換, 149  
文字の大きさ, 260  
文字を修飾する, 214  
モデレータ, 144  
矢印, 278  
ユーザ ID, 15  
ヨーロッパ系の記号, 276  
リージョン, 193  
リスト (箇条書き), 215  
リダイレクション, 184  
リダイレクト, 373  
リンク, 205, 222  
ルートディレクトリ, 101  
ルートメニュー, 37  
ルビ, 260  
レターモード, 123  
ローカルニュースグループ, 133  
枠, 265

## 配布、改変

このドキュメントは非営利目的に利用する限り、自由に複写、改変、配布して構いません。逆に営利目的に利用する事は許しません。この基本線を守る限り、あなたはこのドキュメントに対して何をしても自由です。ここではこれ以上述べません。あなたが執筆者達の期待を裏切らないよう、信じています。

## 連絡など

訂正や誤りに対する連絡は京都産業大学の計算機センター教育研究システム課までお願いします。ここはまずい、こうした方が良くという相談は大歓迎です。我々の環境は時代に合わせてどんどん変化します。このドキュメントもまずいところはどんどん直して変えていって欲しいと思います。そうしてこのドキュメントが実際に役に立つものとして成熟して行けば良いなと考えているのです。

### 執筆者

谷川 正幸 竹内 茂夫 大本 英徹 安田 豊  
山崎 英知 辻本 将彦 笠 克明 松浦 正和  
坪内 伸夫 吉田 浩史 重田 裕之 土肥 順一  
岡田 光博 開原 潮 尾崎 孝治

### 発行

京都産業大学 計算機センター教育研究システム課  
〒 603-8555 京都市北区上賀茂本山  
電話 075-705-1483

## 謝辞

このガイドでは計算機センターが、コンピュータを初めて使うところから電子メールやネットニュースが使えるようになるまでの一連のチュートリアルを書いています。残りの部分については cc 環境利用者の有志から原稿を頂きました。

まず最近話題の WWW では、HTML おいしいホームページの作り方を外国語学部の竹内茂夫先生に書いていただきました。Mule の章とそのリファレンス、FAQ の部分を当時理学部計算機科学科の学生だった尾崎君が書いてくれています。そして彼は現在計算機センター職員となりこのガイドの編集に携わっています。また、FAQ は計算機センターの MiCS 補助員（計算機運用補助員）がまとめてくれたものを利用しています。著作権法の部分は計算機センターの開原君がまとめてくれています。UNIX コマンドリファレンスは元計算機センターの安田君<sup>7</sup> が cc 環境の一ユーザとして書いています。ダイヤルアップ IP は安田君、尾崎君と一般教育研究センターの吉田君の合作となっています。ガイド執筆では直接現れませんが、cc 環境の Mule の設定には理学部の立木先生や他の方々から多くの御協力を得て何とかなりました。

また、 $\text{\LaTeX}$  の章を、物理学科の辻本君<sup>8</sup> が書き起こしてくれたものを元に理学研究科物理学専攻の山崎君が、 $\text{\AUCTEX}$  の章については理学研究科物理学専攻の松浦君<sup>9</sup> がそれぞれ書いてくださいました。

本紙に以前掲載した Mathematica の章については理学研究科数学専攻の笠君が、NQS、*xv*、xpaint の章については安田君が、NeXT の章は開原君が、レポートシステムは尾崎君が、GNUPLOT の章は理学部の谷川正幸先生がそれぞれ書いてくださり、現在は Web ページに一部掲載しています。

その他にも多くの方の協力を戴きました。

それら全ての人に感謝いたします。

## 経緯

cc 環境は 1993 年の夏に計算機センターが導入した UNIX マシン群を核としたコンピュータ利用環境です。それまで大学の中に共用 UNIX 環境が存在しなかったという事もあって、当時大学の中では UNIX の利用者は少なかったのですが、1994 年の春から一般学生にも授業と関わりなくアカウントを出すということになりました。そこで授業などとは関係なく UNIX 環境を利用することになる学生のために、ガイドが必要となったのです。そうして「UNIX ガイド」初版が cc 環境の設計者でもあった当時の計算機センター職員安田君によって書かれました。

1994 年の夏には多くの人の執筆協力を得て「UNIX ガイド追補版」を出版する事ができました。これは学内で UNIX 利用者がそれだけ育った事を象徴する出来事でした。1995 年この二冊を合本して更に内容を加えた第二版が出版されました。1996 年に「UNIX ガイド」はさらに内容を加えて「コンピュータガイド・インターネット編」と「コンピュータガイド・アプリケーション編」に分冊されることになりました。第二版がすでに 400 ページ近い厚さになっており、常に携帯してもらうために、もう少し見やすく軽くなれないかとの配慮からです。1997 年は cc 環境の変化に伴い内容を更新しました。1998 年は、ネットワークが使えるようになるまでをターゲットに絞り込んでインターネット編を構成し、アプリケーション編を Web 化しました。

そしてこの度、より洗練された内容をめざしてインターネット編を作成する運びとなりました。

Netscape に代表されるブラウザは簡単な操作で WWW の利用を可能にし、利用者層を拡大しました。いわばインターネットの「大衆化」を急激に進めたといえます。いまや 4 回生が就職における求人情報収集のためにインターネットを使うことが当然のように行われています。1993 年の cc 環境設計当時には想

---

<sup>7</sup> 安田 豊君は 1995 年 11 月に京都産業大学を退職し神戸大学経済経営研究所に移られました。その後も cc 環境のために協力してくれています。

<sup>8</sup> 辻本君は 1995 年 3 月卒業

<sup>9</sup> 松浦君は 1997 年 3 月修了

像できなかったほど、大衆化は進んでいます。「コンピュータガイド・インターネット編」はUNIXをあまり意識せずにインターネットを使う人達のために書かれています。

cc環境の設計者である安田君は、多くの利用者によって支えられている現在のcc環境をみて「1993年に苦労してcc環境の設計を行っていたのが嘘のように昔の事に思える。最初の半年の利用者には何のドキュメントも用意してやれず苦労を掛けたと今更ながらに思う。」と述懐しています。

これからcc環境がどのように育っていくのか予想がつかない部分もありますが、このドキュメントがその助けになってくれればと願っています。

最後に。論文やレポート時代の貴重な時間をさいて、この「コンピュータガイド」を書いてくださった有志の方々、そしてその成果物のとりまとめにまるまる一カ月を費やした編集者たち、こうしてcc環境を支えてくれているすべての人々に感謝します。

1999.4.8 計算機センター教育研究システム課