

UNIX入門 & 電子メール入門

UNIX入門と電子メール入門は、1994年10月に情報処理センター主催で開催した初心者を対象にした講習会で使用したテキストです。

今回の広報に掲載するにあたってテキストの修正及び一部内容を追加しています。

【UNIX入門】

情報処理センター 小野 隆久

onot@cc.saga-u.ac.jp

【電子メール入門】

情報処理センター 江藤 博文

etoh@cc.saga-u.ac.jp

UNIX (ワークステーション) 入門

1 UNIXの歴史

UNIXは、1969年に米国のAT&T社のベル研究所で開発されたOS (Operating System) です。UNIXという名前は、UNI (単一の) という意味で付けられており、UNIXは軽くすっきりとまとまったOSを目標にして開発されました。

また、1975年には、UNIXのソースプログラムが一般公開され、AT&T社と契約を結んで、UNIXを自由に改良することができるようになりました。その後、UNIXは大きく2つに分かれて成長することになり、その一つがカリフォルニア州立大学バークレイ校で開発されたバークレー版UNIXである★BSDともう一つはAT&T社のUNIXであるSystem Vです。

★BSD版UNIXは、主に大学や研究所を中心に使用され、System V版UNIXは、コンピュータ・メーカーを中心として主に商用機種に使用されていますが、現在、この二つのUNIXとその他のUNIXの統一 (標準化) の動きが出ています。

UNIXは、主にワークステーション (EWS) に搭載されていますが、最近では、★パソコン版のUNIXも出てきています。

2 UNIXの特徴

UNIXは、コンピュータを管理するための専用のOSで、パソコンのMS-DOSやSystem 7などと同じOSの一つです。

UNIXは、次のような特徴を持っています。

(1) 汎用のOS

(2) ★マルチユーザーのOS

複数の利用者が同時に一台のコンピュータを利用することができる。そのため、複数のコンピュータ及び端末を接続して利用できる。

(3) ★マルチタスクのOS

複数の仕事を同時に行っているように処理することができる。

(4) 会話型のOS

利用者がキーボードからコマンドを入力するとコンピュータからの応答がディスプレイに表示される。その繰り返しにより処理を行う。

(5) ★階層型のファイルシステムのOS

UNIXでは、ファイルを分類・整理して管理しやすいように、会社の組織などによく似た階層構造を採用しています。

(6) 入出力がシンプルで統一されたOS

(7) 便利なコマンドが数多く用意されているOS

3 情報処理センターの研究用EWSについて

情報処理センターの研究用システムは、複数台のEWSとパソコンをネットワークで接続しており、特に、EWSは、学内LANに接続されているパソコンから利用することができます。

現在、研究用として利用できるEWSは下記のようになっています。

(1) ★E-Mail (電子メール) の利用

SunSS10 : himiko (133.49.4.3)

(2) ★ftp (ファイル転送) の利用

SunSS10 : sagagw (133.49.4.1)

(3) 科学技術計算 (Fortran, Pascal, C) 用EWS

SunSS10 : imari (133.49.4.9)

SunSS10 : arita (133.49.4.11)

SunClassic : ogi (133.49.4.12)

SunClassic : kasima (133.49.4.13)

SunClassic : tosu (133.49.4.14)

★BSD

Barkley Software Distributionの略です。

★BSD版UNIX

センターの研究用EWS (一部のEWSを除く) 及び教育用EWSのOSとして使用しています。

★パソコン版のUNIX

PC98系やDOS/V機では、BSD/386やPANIXなどがあり、Macでは、A/UXがあります。

★マルチユーザーのOS

パソコンのOSは、一般にシングルユーザーのOSです。

★マルチタスクのOS

パソコンのOSは、一般にシングルタスクのOSです。

★階層型のファイルシステム

パソコンのOSも、階層型 (ディレクトリ) ファイルシステムを採用しています。

★E-Mail

EWSによる電子メールの利用。学内LANに直結したパソコンでメール送受信ソフトを使用すると直接パソコンからメールが送受信ができます。

★ftp

EWS間によるファイル転送。学内LANに直結したパソコンとEWS間でも利用できます。

4 EWSのユーザー名の取得

EWSのOSであるUNIXはマルチユーザ環境ですので、必ず管理者を通してユーザー名を取得しなければ利用することはできません。

情報処理センターのEWSを利用する場合は、ユーザー登録のための申請書を提出しユーザー名を取得して下さい。このユーザー名は、himiko、arita、imari、ogi、kasima、tosuの各EWSで共通で使用することができます。

5 学内LANに接続されたパソコンの接続手順

研究室のパソコンから情報処理センターのEWSを利用する場合は、学内LANの接続方法によって接続手順が異なります。

(1) ★C・S経由接続のパソコン

- ★エミュレータ・ソフトを起動し、C・Sと接続します。
- EWSに接続します。[telnet ホスト名 と入力します]
- EWSにログインします。
- 処理 (E-Mail等) をします。
- EWSからログアウトします。
- エミュレータ・ソフトを終了します。

(2) 学内LAN直結のパソコン

- ★エミュレータ・ソフトを起動し、EWSに接続します。
- EWSにログインします。
- 処理 (メール等) をします。
- EWSからログアウトします。
- エミュレータ・ソフトを終了します。

ここでは、接続手順については詳しくは説明しません。

6 EWSのログイン、ログアウト

6.1 EWSにログインする前に

UNIXでは、入力する文字は、大文字と小文字を区別します。そのため、文字を入力する場合は、CAPSキーをオフにし大文字を入力するときは、SHIFTキーを押しながら入力するクセをつけるようにしましょう。特にユーザー名やパスワードを入力する場合は、注意しましょう。

6.2 EWSにログイン

パソコンをEWSに接続するとEWSはログイン待ち状態になります。ここで、ユーザー名を入力し、パスワードを入力します。

```
4.3 BSD UNIX(himiko) (ttyxx)
```

```
login:ユーザー名
```

```
Password:パスワード (画面には表示されません。)
```

```
Last login: Mon Sep 26 12:51:02 from comp3
```

```
SunOS Release 4.1.2-JLE1.1.2 (HIMIKO-slip) #2: Wed Apr 6 16:18:00 JST 1994
```

```
ユーザー名@himiko[1]
```

ユーザー名とパスワードを正確に入力するとプロンプト (ユーザー名@himiko[1]) が表示されます。

(1) パスワードについて

UNIXにおけるパスワードは、各利用者が管理するものですから管理者にも利用者のパスワードを知ることはできません。そのため、利用者がパスワードを忘れた場合は、EWSを利用できなくなりますので、利用者はパスワードの管理には注意しましょう。

また、パスワードの初期設定は、利用者がログインしやすいように簡単なパスワードにしています。そのため、別の利用者が勝手にログインできる確率が高くセキュリティ上問題があります。ですから、各利用者は最初のログイン時に、必ず★パスワードを変更するようにしましょう。これは、利用者がコンピュータを利用する上での責任でもあります。

パスワードを変更する場合は、次の事に注意してパスワードを付けるよ

★C・S

コミュニケーション・サーバーの略です。この装置は、学内LANに接続されており、各研究室のパソコンとRS232Cケーブルで接続されています。そのため、通信速度は、9600bpsです。

★エミュレータ・ソフト

(1) C・S経由接続のパソコン

PC98系ではHTerm。Macでは、Ninja-Term等があります。

(2) 学内LAN直結のパソコン

PC98系では、TCP/IPボードを購入する時に一緒に購入して下さい。Macでは、NCSATelnet-J (フリーウェア) 等があります。

★MS-DOSは

MS-DOSはシングルユーザーのOSですので、パソコンを利用する場合、ログイン、ログアウトなどという面倒な手順は必要がありません。

★EWSにログイン

EWSにログインすると、プロンプトが表示される前に、メールが来ていると

```
You have new mail.
```

というメッセージが表示されます。

ユーザーは、このメッセージを確認したら、必ずメールを読むようにしましょう。

★パスワードを変更

パスワードの変更は、自分のファイルを守るためだけでなく、利用しているEWSのシステムを守るということも自覚しましょう。

うにしましょう。

- a. 英字で始めます。
- b. 8文字以上にします。(但し、8文字以降はパスワードの対象外)
- c. 単語をつけないようにしましょう。
- d. 特殊文字を入れましょう。(%,&,#,!,?,-等は使用可能)
- e. 英字、数字、上記の特殊文字以外は使用不可能です。
- f. 大文字、小文字の区別をします。
- g. 人に教えず、自分で忘れないようにします。

例) nabe%sima Ichilro HON?jyo

(2) パスワードの変更手順

パスワードを変更する場合は、★yppasswdコマンドで変更します。変更手順は、次のような手順で行います。

```
ユーザー名@himiko[1] yppasswd
Changing NIS passwd for user on himiko
Old passwd 現在のパスワード (画面には表示されません)
New passwd 新しいパスワード (画面には表示されません)
Retry new passwd 新しいパスワード (画面には表示されません)
NIS entry changed on himiko
ユーザー名@himiko[1]
```

パスワードの変更が間違っていなければNIS entry changed on himikoというメッセージがでて終了します。もし、間違えたら、再度yppasswdコマンドを実行して下さい。

新しいパスワードは、次のログイン時から有効となります。

6.3 EWSからログアウト

EWSの利用が終了したなら、必ずUNIXに処理の終了を知らせるためにログアウトを行わなければなりません。

ログアウトの手順は、次のように行います。

```
ユーザー名@himiko[1] logout
```

C・S経由でEWSに接続していれば、★C・Sのプロンプトが画面に表示されます。

6.4 パニックの脱出方法

UNIXを使用する上で入力したコマンドの実行が止まらない。キー入力が出来ない。文字を入力しても何も応答がない。といったパニックにみまわれると初心者の方はどう対処したらいいかわからなくなることがあります。

パニックになった場合は、次のように対応してみましょう。

- (1) 文字の入力を終了させてみましょう。
CTRL-Dキーを押してみましょう。
- (2) 1の操作で駄目な場合は、コマンドを強制終了させてみましょう。
CTRL-Cキーを押してみましょう。
- (3) コマンドの強制終了がきかない場合は、一時的に停止させてから強制終了させてみましょう。
CTRL-Zキーを押して一時停止させると、プロンプトが表示されます。
その後、kill %%を入力し強制終了させます。強制終了したというメッセージが表示されたらOKです。
もしメッセージがでない場合は、
kill -KILL %%と入力してみましょう。

上記の操作を行ってもパニックを回避できない場合は、管理者に連絡してみましょう。

★文字の入力ミスをした場合

例えば、mkdirと入力しようとして、mkdarと入力した場合は、通常バックスペースでカーソルを戻して修正しますが、キーボードのBSキーが使用できない場合があります。そのときは、

CTRL-Hキー

でカーソルを戻して修正します。

CTRLキーとHキーの間にある-というのは、CTRLキーを押したままHキーを押すというキー操作を意味します。

UNIXの利用では、これから先、このようなキー操作が随所に出てきます。

★yppasswdコマンド

yppasswdコマンドは、LANで接続されている複数のEWSが一台のEWSで管理されている場合にパスワードを変更するためのコマンドです。

また、一台のEWSだけで他のEWSを管理していない場合は、passwdコマンドでパスワードを変更します。

★C・Sのプロンプト

例えば、ECS*** (9) > と表示されます。***は接続しているC・Sによって異なります。

★MS-DOSは

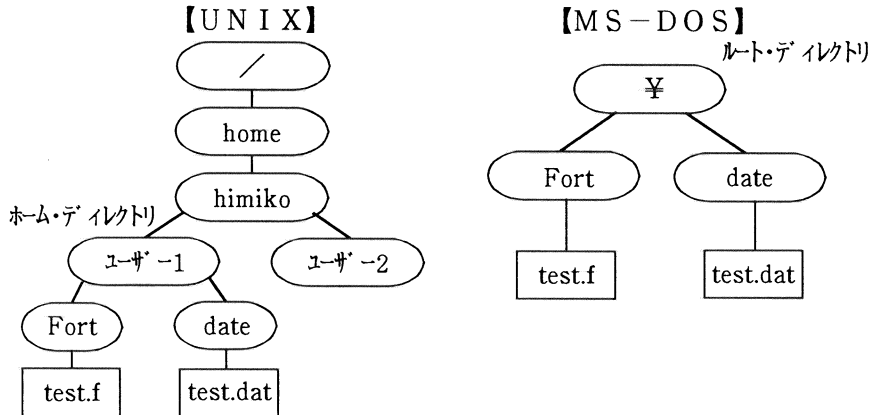
パソコン (MS-DOS) で、パニックになった場合は、パソコンをリセットしたり電源を切ったりして、パニックを回避することができますが、EWS (UNIX) では、たくさんのユーザーが使っていますので、パニックになったからといってEWSの電源を簡単に切ったりすることはできません。

7 MS-DOS利用者のためのUNIX入門

この章では、PC98 (MS-DOS) を利用しているユーザーを対象に、UNIXのコマンドに対応するMS-DOSのコマンドを交えながらUNIXコマンドを紹介していきます。

7.1 ファイルの構造

UNIXのファイル構造は、MS-DOSのファイル構造と同じ、階層的ファイル構造になっています。



★UNIXでは

ユーザー1が、ホーム・ディレクトリとなります。

★MS-DOSでは

¥が、ルート・ディレクトリとなります。

7.2 ディレクトリの管理

(1) ログイン時のディレクトリの位置

UNIXの特徴で紹介したようにUNIXはマルチユーザーのOSです。もし、多人数のユーザーが利用した場合、一つのディレクトリにファイルを作成してしまうと他のユーザーのファイルを勝手に消去したり、内容を変更したりと大混乱を起し管理が出来なくなってしまいます。

そのため、管理者は、ユーザー登録を行うときに各ユーザーごとにディレクトリを作成し、ユーザーがログインすると必ずそのディレクトリの中に入るようになっています。ですから、ユーザーは、そのディレクトリの中でファイルを管理するようになるわけです。

(2) ワーキング・ディレクトリ [pwd]

ユーザーがログインすると必ず管理者が作成したディレクトリの中にいます。このログイン時のディレクトリをホーム・ディレクトリ (MS-DOSではルート・ディレクトリ) といいます。

このホーム・ディレクトリを調べるコマンドが、pwdコマンドです。

```
ユーザー名@himiko pwd
/home/himiko/ユーザー名 ..... ホーム・ディレクトリ
ユーザー名@himiko
```

pwdコマンドは、Print Working Directoryの略でつけられたコマンドで、ワーキング・ディレクトリ (作業ディレクトリ) の位置を表示します。ログインしてすぐにpwdコマンドを実行すると自分の★ホーム・ディレクトリが表示され、別のディレクトリに移動した後にpwdコマンドを実行すると移動後のワーキング・ディレクトリを表示します。

(3) ディレクトリの作成 [mkdir ディレクトリ名]

新しいディレクトリを作成するコマンドが、mkdirコマンドです。ユーザーが作成するファイルは、すべて現在いるワーキング・ディレクトリの中に作成されます。

ファイルを管理する場合は、ファイルの内容に応じてディレクトリごとに分けるとファイルの管理もたいへん楽になります。

例えば、ワーキング・ディレクトリ (ホーム・ディレクトリ) の下にFortディレクトリを作成する場合は、次のように入力します。

```
ユーザー名@himiko mkdir Fort
ユーザー名@himiko
```

★MS-DOSは

MS-DOSはシングルユーザーのOSですので、パソコンを利用する場合、ログインなどという面倒な手順は必要がありません。

しかし、パソコンの電源を入れれば、誰でも勝手にファイルをいじることができるという管理上、大きな欠点を持っているわけです。

★MS-DOSでは

```
A:¥> cd
A:¥ ..... カレント・ディレクトリ
A:¥>
```

★ホーム・ディレクトリ

ホーム・ディレクトリで作業を行えば、ホーム・ディレクトリがワーキング・ディレクトリとなるわけです。

★MS-DOSでは

```
A:¥> mkdir Fort
A:¥>
```

これで、ワーキング・ディレクトリ (/home/himiko/ユーザー名) の下に Fort というディレクトリが作成されました。

(4) ディレクトリの移動 [cd ディレクトリ名]

ディレクトリを移動するコマンドが、cd コマンドです。例えば、Fort ディレクトリに移動する場合は、次のように入力します。

```
ユーザー名@himiko cd Fort
ユーザー名@himiko pwd
/home/himiko/ユーザー名/Fort
ユーザー名@himiko
```

これで、Fort ディレクトリにワーキング・ディレクトリが移動しました。では、ホーム・ディレクトリに戻ってみましょう。ホーム・ディレクトリに戻る場合は、ディレクトリ名を指定せずに cd コマンドを実行します。

```
ユーザー名@himiko cd
ユーザー名@himiko pwd
/home/himiko/ユーザー名
ユーザー名@himiko
```

次に、ホーム・ディレクトリの下に data ディレクトリを作成し、data ディレクトリに移動してみましょう。

```
ユーザー名@himiko mkdir data
ユーザー名@himiko cd data
ユーザー名@himiko pwd
/home/himiko/ユーザー名/data
ユーザー名@himiko
```

今、ホーム・ディレクトリの下には、Fort と data という二つのディレクトリが作られたこととなります。

階層構造のファイル・システムを採用しているコンピュータを使用する場合、ディレクトリ間をいかに速く渡り歩けるかが大きなポイントとなります。

これから、ディレクトリを渡り歩いてみることにしましょう。現在、ホーム・ディレクトリの下に data ディレクトリにいますので、ひとつ上のディレクトリ、つまりホーム・ディレクトリに移動してみましょう。

```
ユーザー名@himiko cd ..
ユーザー名@himiko pwd
/home/himiko/ユーザー名
ユーザー名@himiko
```

このように一つ上のディレクトリに移動する場合は、ディレクトリ名の入力のところで .. (ピリオドを2回入力) と入力します。二つ上のディレクトリに移動する場合は、../.. と入力します。

次に、Fort ディレクトリに移動した後、直接 data ディレクトリに移動してみましょう。

```
ユーザー名@himiko cd Fort
ユーザー名@himiko pwd
/home/himiko/ユーザー名/Fort
ユーザー名@himiko cd ../data
ユーザー名@himiko pwd
/home/himiko/ユーザー名/data
ユーザー名@himiko
```

★MS-DOSでは

```
A:¥> cd Fort
A:¥Fort> cd
A:¥Fort
A:¥Fort>
```

MS-DOSのcdコマンドは、chdir と入力してもよい。

★MS-DOSでは

```
A:¥Fort> cd ¥
A:¥> cd
A:¥
A:¥>
```

★MS-DOSでは

```
A:¥> mkdir data
A:¥> cd data
A:¥data> cd
A:¥data
A:¥data>
```

★MS-DOSでは

```
A:¥> cd ..
A:¥> cd
A:¥
A:¥>
```

MS-DOSでも一つ上のディレクトリに移動する場合に、.. を使用します。

★MS-DOSでは

```
A:¥> cd Fort
A:¥Fort> cd
A:¥Fort
A:¥Fort> cd ¥data
A:¥data> cd
A:¥data
A:¥data>
```

どうですか、上手にディレクトリを移動することができましたか。

(5) 簡単なディレクトリの移動 [`pushd {ディレクトリ名}`, `popd`]

`cd`コマンドよりもっと簡単にディレクトリを移動することができるコマンドが、`pushd`コマンドと`popd`コマンドです。

この`pushd`コマンドと`popd`コマンドは、主に一時的にディレクトリを移動し、直ぐに元のディレクトリに戻りたい場合に使用します。

例えば、ホーム・ディレクトリからFortディレクトリに移動し、再度ホーム・ディレクトリに戻りたい場合は、次のように入力します。

```
ユーザー名@himiko pushd Fort ... Fortディレクトリに移動します。
ユーザー名@himiko pwd
/home/himiko/ユーザー名/Fort
ユーザー名@himiko popd ... ホーム・ディレクトリに戻ります。
ユーザー名@himiko pwd
/home/himiko/ユーザー名
ユーザー名@himiko
```

このように`cd`コマンドや`pushd`、`popd`コマンドを使用して、あちこちのディレクトリを渡り歩けるようになれば、少しでもUNIXの世界が広がるでしょう。

(6) ディレクトリの削除 [`rmdir` ディレクトリ名]

不要になったディレクトリを削除するコマンドが、`rmdir`です。

例えば、不要になったdataディレクトリを削除する場合は、次のように入力します。

```
ユーザー名@himiko cd
ユーザー名@himiko pwd
/home/himiko/ユーザー名
ユーザー名@himiko rmdir data
ユーザー名@himiko
```

`rmdir`コマンドで削除できるディレクトリは、ディレクトリの下にファイルがない場合です。一つでもディレクトリの下にファイルが存在する場合は、ディレクトリを削除することはできません。

また、`rmdir`コマンドは、複数のディレクトリを一度に削除することもできます。例えば、Fortとdataディレクトリを一度に削除する場合は、次のように入力します。

```
ユーザー名@himiko mkdir data
ユーザー名@himiko rmdir Fort data
ユーザー名@himiko
```

(7) ディレクトリとファイルを一気に削除 [`rm -r` ディレクトリ名]

ディレクトリの下にファイルがあるディレクトリをファイルと一緒に削除するコマンドが、`rm`コマンドです。

例えば、Fortディレクトリとその下にあるファイルを一緒に削除する場合は、次のように入力します。

```
ユーザー名@himiko rm -r Fort
rm: override protection 444 for Fort/t.dat? y .. 書き込み禁止のファイル
ユーザー名@himiko                               を削除します。
```

この`rm`コマンドを実行する場合は、ディレクトリ以降のファイルまで削除してしまいますので、大事なファイルまで削除しないように十分に注意が必要です。ただし、書き込み禁止のファイルについては、削除してもよいかの確認が行われます。

また、書き込み禁止のファイルまで確認なしに削除する場合は、次のように入力します。

★MS-DOSでは

MS-DOSでは、UNIXの`pushd`、`popd`コマンドの機能に相当するコマンドはありません。

★MS-DOSでは

```
A:¥> cd ¥
A:¥> cd
A:¥
A:¥> rmdir data
A:¥>
```

MS-DOSの`rmdir`コマンドは、`rd`という省略形でも良い。

★MS-DOSでは

MS-DOSの`rmdir`コマンドでは、二つ以上のディレクトリを一度に削除することはできません。

`n`と入力すると書き込み禁止のファイルは削除されず、Fortディレクトリも削除されません。

```
ユーザー名@himiko rm -rf Fort
ユーザー名@himiko
```

このコマンドを実行すると一気にディレクトリごとファイルまで削除してしまいますので、できるだけこのコマンドは使用しないようにしましょう。

(8) パスの指定方法

UNIXとMS-DOSは、同じ階層型のファイル構造ですので、ディレクトリの下にあるファイルを指定するパスの指定方法は同じです。

目的のファイル名までのディレクトリを、UNIXの場合は"/"、MS-DOSの場合は"¥"で区切っていきます。

例えば、Fortディレクトリの下でのtest.f77のパス名は、次のように書くこととなります。

【UNIXの場合】

```
Fort/test.f77
```

【MS-DOSの場合】

```
Fort¥test.f77
```

★MS-DOSでは

```
A:¥> del Fort
ディレクトリ内のすべてのファイル
が削除されます！
よろしいですか(Y/N) y
A:¥> rmdir Fort
A:¥>
```

★UNIXのコマンドには

UNIXのコマンドには、いくつかのオプションが用意されています。

```
rm -r ディレクトリ名
オプション
```

このオプションをつけると、コマンドの機能がさらに向上します。これからも、オプションをつけたコマンドが数多く紹介されます。

7.3 ファイルの管理

コンピュータに何らかの処理をさせたい場合、ユーザーは、ファイルの作成、修正、削除、実行といった作業にほとんどの時間を費やします。

そのため、ユーザーは、いかにファイルを自由に扱うことができるかによって、作業時間を短縮することが可能となります。

(1) 簡単なファイルの作成 [cat > ファイル名]

数行の内容のファイルを作成することができるコマンドが、catコマンドです。

例えば、所属と氏名の内容のファイルtest.datを作成する場合は、次のように入力します。

```
ユーザー名@himiko cat > test.dat
SAGA UNIVERSITY
Adachi Yumi
```

CTRL-Dキーで終了します。

```
ユーザー名@himiko
```

(2) MS-DOSからEWSのファイル転送

catコマンドと★HTerm (通信ソフト) の機能を使用してMS-DOSのテキスト・ファイルをUNIXのファイルに転送することができます。

MS-DOSのファイル転送の手順

[1] catコマンドで新規ファイルを開きます。

```
ユーザー名@himiko cat > test.dat
```

[2] COPYキーを押してHTERMのメニュー画面を開きます。

[3] File Xferにカーソルを移動させ、RETURNキーを押します。

[4] File up load offにカーソルを移動させRETURNキーを押します。

[5] そうすると画面下にUp load file name:が表示されますので、MS-DOSファイル名を入力しRETURNキーを押します。

[6] MS-DOSのファイルの内容が画面上に流れていきます。

[7] MS-DOSのファイルの内容がすべて表示されたら、RETURNキーを押した後、CTRL-Dキーを押します。

以上の手順でMS-DOSのファイルをUNIXのファイルに転送することができます。

(3) ファイルの一覧を表示 [ls {オプション} {ファイル名}]

ファイルの一覧を表示するコマンドが、lsコマンドです。

例えば、ホーム・ディレクトリの下にあるファイルの一覧を表示させたい場合は、次のように入力します。

```
ユーザー名@himiko cd
ユーザー名@himiko ls
test.dat
ユーザー名@himiko
```

また、lsコマンドにオプションをつけるとさらに詳しいファイルの情報が得られます。

ファイルの情報まで表示させたい場合は、次のように入力します。

```
ユーザー名@himiko ls -l
total 1
-rw-r--r-- 1 ユーザー名 28 Oct 5 16:55 test.dat
ユーザー名@himiko
```

ファイルの情報の項目は、下記のようにになっています。

★MS-DOSでは

```
A:¥> type con > test.dat
SAGA UNIVERSITY
Adachi Yumi
```

STOPキーを押す

```
A:¥>
```

★HTerm

HTermは、PC98をC・S接続している場合に、使用する通信ソフトです。このソフトは、フリーウェアですので、センターにもあります。

MacをC・S接続している場合は、Ninja-Term等の通信ソフトを使用して接続します。また、catコマンドでファイルを開いた後に、コピー&ペーストの機能を使用すれば簡単にファイル転送ができます。

{ } 内は省略可能

★MS-DOSでは

```
A:¥> dir/w
test.dat
A:¥>
```

★MS-DOSでは

```
A:¥> dir
test dat 28 94-10-13 10:00
1個 28バイトのファイル....
A:¥>
```

```
rw-r--r-- 1 ユーザー名 28 Oct 5 16:55 test.dat
(1) (2) (3) (4) (5) (6) (7)
```

- (1)★ファイルの種類 (-,d)
- (2)アクセスの許可情報
- (3)リンク数
- (4)ファイルの所有者
- (5)ファイルのバイト数
- (6)ファイルを最後に更新したときの日時
- (7)ファイル名

各ユーザーごとにファイル名の先頭に、(ピリオド)がついたファイルがいくつかあります。これは、ログオン時の環境を設定するための★利用者用環境設定ファイルです。

これらのファイル名も表示させたい場合は、次のように入力します。

```
ユーザー名@himiko ls -a
```

ディレクトリと実行ファイルがわかるように表示させたい場合は、次のように入力します。

```
ユーザー名@himiko ls -F
```

ファイルがディレクトリの場合は” / ”、実行ファイルの場合は” * ”をファイル名の後につけて表示します。

ファイル名をアルファベット順に表示させたい場合は、次のように入力します。

```
ユーザー名@himiko ls -r
```

(4) ファイルの種類を調べる [file ファイル名]

ファイルの種類を調べるコマンドが、fileコマンドです。

例えば、test.datというファイルがどんなファイルかを調べたい場合は、次のように入力します。

```
ユーザー名@himiko file test.dat
```

```
test.dat ascii text
```

```
ユーザー名@himiko
```

ファイルの種類には、主に次のようなものがあります。

```
ascii text      : アスキーコードのテキストファイル
c program text  : C言語のプログラム
EUC text       : EUCコードのテキストファイル
C-shell commands : シェルコマンドファイル
```

fileコマンドの結果からtest.datというファイルは、アスキーコードのテキストファイルということがわかります。

(5) ファイルの内容を表示する [cat {オプション} ファイル名]

ファイルの内容を画面に表示するコマンドが、catコマンドです。

例えば、test.datの内容を画面に表示させたい場合は、次のように入力します。

```
ユーザー名@himiko cat test.dat
```

```
SAGA UNIVERSITY
```

```
Adachi Yumi
```

```
ユーザー名@himiko
```

★ファイルの種類

- : 一般のファイル
d : ディレクトリ

★利用者用環境設定ファイル

MS-DOSでいえばconfig.sysに相当するファイルです。利用者用環境設定ファイルにはいろんなファイルがありますが、主に次のようなファイルがあります。

```
.cshrc
.login
.logout
```

上記のファイルをエディタなどで編集し、利用者個人で使用しやすい環境に設定することができます。

★MS-DOSでは

```
A:¥> dir/o
test dat 28 94-10-13 10:00
A:¥>
```

★MS-DOSでは

MS-DOSでは、UNIXのfileコマンドの機能に相当するコマンドはありません。

★MS-DOSでは

```
A:¥> type test.dat
SAGA UNIVERSITY
Adachi Yumi
A:¥>
```

ファイルの内容を表示するときに行番号をつけて表示させたい場合は、次のように入力します。

```
ユーザー名@himiko cat -n test.dat
1 SAGA UNIVERSITY
2 Adachi Yumi
ユーザー名@himiko
```

ファイルの内容を表示するとき空白行には行番号をつけずに行番号つきで表示させる場合は、次のように入力します。

```
ユーザー名@himiko cat -b test.dat
1 SAGA UNIVERSITY

2 Adachi Yumi
ユーザー名@himiko
```

(6) ファイルの内容をページごとに表示する [more (オプション) ファイル名]
行数が多いファイルの内容をページごとに表示させるコマンドが、moreコマンドです。

例えば、test.datの内容をページごとに画面に表示させたい場合は、次のように入力します。

```
ユーザー名@himiko more test.dat
SAGA UNIVERSITY
Adachi Yumi
ユーザー名@himiko
```

上記の場合は、ファイルの行数が少ないのでmoreコマンドの機能が役にたっていませんが、行数の多いファイルの場合であればページごとに内容が表示され、RETURNキーを押すと1行ごとにスクロールアップし、SPACEキーを押すと次のページに移ります。

moreコマンドは、指定した行数単位で表示をさせることもできます。例えば、test.datの内容1行ごとに表示させたい場合は、次のように入力します。

```
ユーザー名@himiko more -1 test.dat
SAGA UNIVERSITY
--More-- SPACEキーを押します
Adachi Yumi
ユーザー名@himiko
```

moreコマンドを実行すると--More--というメッセージが表示され、moreコマンド内のコマンド入力待ち状態になります。ここで、コマンドを入力することによっていろんな操作ができます。

- RETURNキー : 1行ごとにスクロールアップします。
- 行数RETURNキー : 指定した行数ごとにスクロールアップします。
- SPACEキー : 次のページに移ります。
- 行数SPACEキー : 指定した行数単位でページを移ります。
- b : 前のページに戻ります。
- 行数b : 指定した行数単位で前のページに戻ります。
- q または Q : 途中でmoreコマンドを終了させます。
- h : ヘルプ機能です。

moreコマンドは、他のコマンドと組み合わせて使用することができます。例えば、ファイル数が多くゆっくりファイルを確認しながらファイルの一覧をページごとに表示させたい場合は、次のように入力します。

```
ユーザー名@himiko ls -C | more
```

★MS-DOSでは

MS-DOSのtypeコマンドには、行番号をつけて表示する機能はありません。

★MS-DOSでは

MS-DOSのtypeコマンドには、空白行には行番号をつけずに行番号をつけて表示する機能はありません。

★MS-DOSでは

```
A:¥> type test.dat | more
SAGA UNIVERSITY
Adachi Yumi
A:¥>
```

1画面表示すると次のようなメッセージが表示されます。

--続きがありますのでどれかキーを押してください--

ここでSPACEキーまたはRETURNキーを押すと次の1画面を表示します。

途中で終了させたい場合は、CTRL-Cキーを押します。

★コマンドの接続機能

UNIXでは、いくつかのコマンドをつないで(パイプライン機能)実行することができます。

コマンドをつないで実行する場合は、コマンドとコマンドの間に| (立て棒)をはさんで入力します。

こうすると1行に複数のファイル名を表示しながら、ページごとに画面に表示させることができます。

(7) ファイルの内容を先頭から表示する [head {-n} ファイル名]

ファイルの内容をファイルの先頭から10行目までを表示させるコマンドが、headコマンドです。

オプションで行数を指定すると先頭から指定した行数までを表示します。例えば、test.datの内容を先頭から15行目までを表示させたい場合は、次のように入力します。

```
ユーザー名@himiko head -15 test.dat
```

(8) ファイルの内容を最後から表示する [tail {オプション} ファイル名]

ファイルの内容をオプションで指定した行数から最後までを表示させるコマンドが、tailコマンドです。

オプションで、-10と指定するとファイルの最後から10行を表示し、+10と指定するとファイルの10行目からファイルの最後までを表示します。例えば、test.datの最後の10行の内容を表示させたい場合は、次のように入力します。

```
ユーザー名@himiko tail -10 test.dat
```

(9) ファイルのコピー [cp {オプション} 元ファイル名 先ファイル名]

ファイルのコピーをするコマンドが、cpコマンドです。

例えば、test.datの内容をtest2.datにコピーしたい場合は、次のように入力します。

```
ユーザー名@himiko cp test.dat test2.dat
```

また、オプションで-iを指定すると既存のファイルに誤ってコピーしないように上書きの確認を求めてきます。

例えば、誤って既存のtest2.datにtest.datの内容をコピーをしようとし上書きをキャンセルしたい場合は、次のように入力します。

```
ユーザー名@himiko cp -i test.dat test2.dat
overwrite test2.dat? n ..... 上書きをキャンセルします。
ユーザー名@himiko
```

ワーキング・ディレクトリの下すべてのファイルをディレクトリにコピーすることもできます。

例えば、ワーキング・ディレクトリの下ファイルをFortディレクトリにコピーしたい場合は、次のように入力します。

```
ユーザー名@himiko cp * Fort
```

オプションで-rを指定するとディレクトリ間のコピーを行うこともできます。

例えば、Fortディレクトリの下ファイルやディレクトリをdataディレクトリの下にコピーしたい場合は、次のように入力します。

```
ユーザー名@himiko cp -r Fort data
```

(10) ファイル名の変更とファイル名の移動 [mv {オプション} 元ファイル名 先ファイル名]

ファイル名を変更したり別のディレクトリに移動させたりするコマンドが、mvコマンドです。

例えば、test.datのファイル名をtest2.datに変更したい場合は、次のように入力します。

★MS-DOSでは

MS-DOSでは、UNIXのheadコマンドの機能に相当するコマンドはありません。

★MS-DOSでは

MS-DOSでは、UNIXのtailコマンドの機能に相当するコマンドはありません。

★MS-DOSでは

```
A:¥> copy test.dat test2.dat
A:¥>
```

MS-DOSの場合は、上書きチェックの確認を行うことができません。そのため、MS-DOSでは大切なファイルは出来るだけattribコマンドで書き込み禁止にしておいたほうがよいでしょう。

★ワイルド・カード (*)

UNIXでは複数のファイルをまとめて指定できるようにワイルド・カードが用意されています。

例えば、test.datとtest2.datの二つのファイルをまとめて指定する場合は、*.datと指定します。

また、*と指定するとすべてのファイルを指定したことになります。

このワイルド・カードは、MS-DOSにも使用されています。

```
ユーザー名@himiko mv test.dat test2.dat
```

また、オプションで -i を指定すると既存のファイルと同じファイル名にしないように確認を求めてきます。

例えば、誤って既存の test2.dat と同じファイル名に変更しようとし変更をキャンセルしたい場合は、次のように入力します。

```
ユーザー名@himiko mv -i test.dat test2.dat
overwrite test2.dat? n ..... ファイル名の変更をキャンセルします。
ユーザー名@himiko
```

ワーキング・ディレクトリの下のすべてのファイルをディレクトリに移動することもできます。

例えば、ワーキング・ディレクトリの下のファイルを Fort ディレクトリに移動したい場合は、次のように入力します。

```
ユーザー名@himiko mv * Fort
```

ディレクトリ間の移動を行うこともできます。

例えば、Fort ディレクトリの下ファイルやディレクトリを data ディレクトリの下に移動したい場合は、次のように入力します。

```
ユーザー名@himiko mv Fort data
```

(11) ファイルの削除 [rm {オプション} ファイル名]

ファイルの削除をするコマンドが、rm コマンドです。

例えば、test.dat のファイルを削除したい場合は、次のように入力します。

```
ユーザー名@himiko rm test.dat
```

また、オプションで -i を指定すると大事なファイルに誤って削除しないように確認を求めてきます。

例えば、誤って既存の test.dat を削除しようとし削除をキャンセルしたい場合は、次のように入力します。

```
ユーザー名@himiko rm -i test.dat
rm:remove test.dat? n ..... 削除をキャンセルします。
ユーザー名@himiko
```

オプションで -r を指定するとディレクトリの下すべてのファイルを削除することもできます。

例えば、Fort ディレクトリの下すべてのファイルを削除したい場合は、次のように入力します。

```
ユーザー名@himiko rm -r Fort
```

(12) ファイルの保護 [chmod 保護モード ファイル名]

ファイルの保護モードを変更するコマンドが、chmod コマンドです。

UNIX のファイルには、4 種類の相手に対し 4 種類の許可権 (アクセスパーミッション) を設定することができます。

4 種類の相手とは、

- [1] u (user: 所有者)
- [2] g (group: グループ)
- [3] o (others: その他のユーザー)
- [4] a (all: 全員)

★MS-DOSでは

```
A:¥> ren test.dat test2.dat
A:¥>
```

MS-DOSでは、既存するファイルと同じファイル名を指定した場合、ファイル名を変更することはできません。

★MS-DOSでは

```
A:¥> ren dir Fort data
A:¥>
```

★MS-DOSでは

```
A:¥> del test.dat
A:¥>
```

★MS-DOSでは

```
A:¥> del test.dat /p
A:¥TEST.DAT,
削除しますか(Y/N)? y
A:¥>
```

MS-DOSでは、誤って大切なファイルを削除した場合は、undelete コマンドで復活させることができます。

★MS-DOSでは

MS-DOSでも、ファイルを保護することができる attrib コマンドがあります。

このコマンドは、ファイルを書き込み禁止にしたり、通常のコマンドではファイル名を表示させないようにすることができます。

4種類の許可権とは、

[1] r (read: 読み出し)

ファイルの内容を表示したり、別のファイルにコピーすることができます。

[2] w (write: 書き込み)

ファイルの内容を変更することができます。

[3] x (execut: 実行)

ファイルの内容が実行可能なプログラムである場合にのみ意味を持ち、その内容を実行することができます。

[4] - (不許可)

ファイルをアクセスすることができません。

ファイルの保護モードには、いろんな設定がありますが、ここでは大切なファイルを誤って削除しないようにファイルを保護する方法を紹介します。

例えば、test.datを誤って削除しないように書き込み禁止の保護をする場合は、次のように入力します。

```
ユーザー名@himiko chmod u-w test.dat
ユーザー名@himiko rm test.dat
rm:remove test.dat? n ... 削除をキャンセルします。
ユーザー名@himiko
```

これは、chmodコマンドでtest.datを書き込み禁止にして、rmコマンドで誤って削除しないようにしているわけです。ただし、ファイルを書き込み禁止している場合は、ファイルの編集はできませんので、ファイルを編集を行いたい場合は、ファイルを書き込みできるようにしなければなりません。

例えば、test.datを書き込みできるようにする場合は、次のように入力します。

```
ユーザー名@himiko chmod u+w test.dat
```

★MS-DOSでは

```
A:¥> attrib +r test.dat
A:¥> del test.dat
アクセスは拒否されました
A:¥>
```

★MS-DOSでは

```
A:¥> attrib -r test.dat
A:¥>
```

7.4 知っていると便利なコマンドの利用

UNIXには200から300種類程のコマンドが提供されており、UNIXを利用する上で、知っておくと便利なコマンドがたくさんあります。また、便利なコマンドの利用方法もあります。

(1) 日時を表示する [date {オプション}]

現在の日時を表示するコマンドが、dateコマンドです。

```
ユーザー名@himiko date
```

また、-u オプションをつけると、グリニッシュ標準時が表示されます。

```
ユーザー名@himiko date -u
```

(2) カレンダーを表示する [cal {月} 年]

カレンダーを表示するコマンドが、calコマンドです。

例えば、1994年10月のカレンダーを表示させたい場合は、次のように入力します。

```
ユーザー名@himiko cal 10 1994
```

(3) ログイン中のユーザーを調べる [who]

ログイン中のユーザーを表示するコマンドが、whoコマンドです。

```
ユーザー名@himiko who
```

(4) コマンドの履歴情報を調べる [history]

入力したコマンドの履歴情報を表示するコマンドが、historyコマンドです。

historyコマンドで表示される入力コマンドの履歴は、利用者用環境設定ファイル：.cshrcにset history=回数という設定値で残しておくコマンドの回数が設定されています。

入力したコマンドの履歴を表示させたい場合は、次のように入力します。

```
ユーザー名@himiko history
```

```
1 who
```

```
2 ls
```

```
3 history
```

```
ユーザー名@himiko
```

また、事前にset history=回数を実行しておくことと残しておくコマンドの回数を再設定することもできます。

例えば、回数を10に再設定したい場合は、次のように入力します。

```
ユーザー名@himiko set history=10
```

```
ユーザー名@himiko history
```

```
1 who
```

```
2 ls
```

```
3 history
```

```
ユーザー名@himiko
```

(5) コマンドの内容をファイルに保存する [コマンド > ファイル名]

入力したコマンドの内容を、ファイルに保存することができます。

例えば、lsコマンドで表示するファイルの一覧をls.txtに保存したい場合は、次のように入力します。

★MS-DOSでは

```
A:¥> date
```

```
現在の日付は1194-10-13.....
```

```
日付を入力して....
```

```
A:¥> time
```

```
現在の時刻は10:00:00.....
```

```
時刻を入力して....
```

```
A:¥>
```

★MS-DOSでは

MS-DOSでは、UNIXのcalコマンドの機能に相当するコマンドはありません。

★MS-DOSでは

```
A:¥> doskey
```

```
A:¥> F・7キーを押します
```

```
1 dir/w
```

```
2 date
```

```
3 time
```

```
A:¥>
```

doskeyコマンドを実行すると入力したコマンドの履歴情報を残します。

```
ユーザー名@himiko ls > ls.txt
```

(6) 2つのファイルをつなぐ [cat ファイル1 ファイル2 > 新規ファイル名]

2つのファイルの内容をつないで、ひとつのファイルに保存することができます。

例えば、test1.datの内容とtest2.datの内容をつないで、test3.datに保存したい場合は、次のように入力します。

```
ユーザー名@himiko cat test1.dat test2.dat > test3.dat
```

(7) ファイルの内容を2つに分ける

[head 回数 ファイル名 > 新規ファイル名, tail 回数 ファイル名 > 新規ファイル名]

ファイルの内容を、2つのファイルに分けることができます。

ファイルの内容を2つのファイルに分けたい場合は、headコマンドとtailコマンドを使って行います。

例えば、test1.datの内容(10行)を先頭行から5行目までをtest2.datに、6行目から最終行までをtest3.datに保存したい場合は、次のように入力します。

```
ユーザー名@himiko head -5 test1.dat > test2.dat
```

```
ユーザー名@himiko tail -5 test1.dat > test3.dat
```

(8) ファイルの内容の真ん中をファイルに保存する

[head 回数 ファイル名 | tail 回数 > 新規ファイル名]

ファイルの内容の真ん中の部分を、ファイルに保存することができます。

例えば、test1.datの内容(10行)の3行目から6行目までをtest2.datに保存したい場合は、次のように入力します。

```
ユーザー名@himiko head -6 test1.dat | tail -3 > test2.dat
```

(9) 別名コマンドの作成と削除

[alias {別名コマンド ' コマンド' }, unalias 別名コマンド]

コマンドを、別名のコマンドに置き換えるコマンドが、aliasコマンドです。これは、ログイン中だけ有効です。

例えば、rm -iコマンドをrmiという別名をつけて実行したい場合は、次のように入力します。

```
ユーザー名@himiko alias rmi 'rm -i'
```

```
ユーザー名@himiko rmi test.dat
```

```
rm:remove test.dat? y ... 削除します。
```

```
ユーザー名@himiko
```

別名コマンドを調べたい場合は、次のように入力します。

```
ユーザー名@himiko alias
```

```
rmi rm -i
```

```
ユーザー名@himiko
```

別名のコマンド(複数指定可)を削除するコマンドが、unaliasコマンドです。

例えば、上記で作成した別名コマンドrmiを削除したい場合は、次のように入力します。

```
ユーザー名@himiko unalias rmi
```

★MS-DOSでは

```
A:¥> dir/w > dir.txt
```

★MS-DOSでは

```
A:¥> copy t1.dat+t2.dat t3.dat
```

★MS-DOSでは

MS-DOSでは、UNIXのファイルの内容を2つに分ける機能を持つコマンドはありません。

★MS-DOSでは

MS-DOSでは、UNIXのファイルの内容の真ん中をファイルに保存する機能を持つコマンドはありません。

★MS-DOSでは

MS-DOSでは、UNIXの別名コマンドの作成と削除する機能を持つコマンドはありません。

(10) オンラインマニュアルの参照 [man {コマンド}]

コマンドの入力方法などを忘れた場合にオンラインマニュアルを参照するコマンドが、manコマンドです。

manコマンドを実行すると、コマンドの入力方法とコマンドの内容について日本語で表示されます。(システムによっては英文の場合もあります。)また、内容の表示は★moreコマンドの機能で表示されます。

例えば、rmコマンドの入力方法を忘れた場合は、次のように入力します。

```
ユーザー名@himiko man rm
```

(11) ログイン中のユーザーにメッセージを送る [write ユーザー名]

ログイン中のユーザーにメッセージを送るコマンドが、writeコマンドです。また、相手もwriteコマンドを実行すると双方向の会話ができるようになります。ログイン中のユーザーは、whoコマンドで調べることができます。

例えば、ログイン中のitiroさんにメッセージを送りたい場合は、次のように入力します。

```
ユーザー名@himiko write itiro
message from itiro ttya...
Hello itiro (o) ..... 一つの会話を終了します。
Good bye (oo) ..... 会話を終了します。
```

CTRL-Dキーで終了します。

```
EOF
ユーザー名@himiko
```

writeコマンドで相手にメッセージを送るとメッセージを受け取った相手の端末の画面が乱れてしまうなど、相手に迷惑をかける場合もあります。writeコマンドを使用する場合は、注意しましょう。

また、writeコマンドを受け付けられないようにするコマンドもあります。そのコマンドがmesgコマンドです。

```
ユーザー名@himiko mesg n ..... メッセージを受け付けません。
ユーザー名@himiko mesg y ..... メッセージを受け付けます。
```

(12) ログイン中のユーザーと会話する [talk ユーザー名]

ログイン中のユーザーと会話するコマンドが、talkコマンドです。

talkコマンドでは、端末の画面を上下2分割し、上段には自分のメッセージ、下段には相手のメッセージが表示されますが、会話の相手もtalkコマンドを実行しなければ、双方向の会話は成立しません。

```
ユーザー名@himiko talk itiro
[Waiting for your party to respond]
```

相手がtalkコマンドを実行すると、次のようなメッセージが表示され会話が開始されます。

```
[Connection established]

自分が入力するメッセージ
-----
相手からのメッセージ
```

★moreコマンド

画面下に表示されるMoreの後に入力するキー。

SPACEキー：次のページを表示
数SPACEキー：指定した数だけ次のページに移動

RETURNキー：1行づつスクロールアップ

数RETURNキー：指定した行数行づつスクロールアップ

b：前のページに戻る

q又はQ：終了する

★MS-DOSでは

MS-DOSでは、UNIXのwriteコマンドの機能に相当するコマンドはありません。

★MS-DOSでは

MS-DOSでは、UNIXのtalkコマンドの機能に相当するコマンドはありません。

相手がいっこうにtalkコマンドを実行しない場合は、次のようなメッセージが表示されます。

```
[Ringing your party again]
```

talkコマンドを終了する場合は、CTRL-Dキーを押します。

(13) コマンドを再実行する [!一度入力したコマンドの頭文字]

ログオン後に一度使用したコマンドを再実行したい場合は、!の後に一度使用したコマンドの頭文字を入力すると短いキー入力で行うことができます。

ただし、historyコマンドで履歴が残っていないと使用することができませんので、historyコマンドで確認しましょう。

例えば、一度lsコマンドを使用した後にもう一度lsコマンドをしたい場合は、次のように入力します。

```
ユーザー名@himiko !!
```

また、!の後にhistoryコマンドで表示された履歴情報の番号でも実行することができます。

例えば、一度lsコマンドを使用した後にもう一度lsコマンドをしたい場合は、次のように入力します。

```
ユーザー名@himiko history
1 who
2 ls
3 history
ユーザー名@himiko !2
```

(14) 他のコンピュータとの接続 [telnet (ホスト名 or IPアドレス)]

インターネット接続されているコンピュータと接続するコマンドが、telnetコマンドです。このコマンドで、他のコンピュータと接続することはできますが、接続したコンピュータにユーザー登録をしていなければ接続したコンピュータを利用することはできません。

例えば、ogi (IPアドレス: 133.49.4.12) というホスト名を持つコンピュータと接続したい場合は、次のように入力します。

[1] ホスト名での接続

```
ユーザー名@himiko telnet ogi
login: ユーザー名 ... キャンセルの場合は、CTRL-Cを押す
Password: パスワード
```

[2] IPアドレスでの接続

```
ユーザー名@himiko telnet 133.49.4.12
```

ログアウトすると自動的にtelnetコマンドも終了します。

(15) 他のコンピュータとのファイル転送 [ftp (ホスト名 or IPアドレス)]

インターネット接続されているコンピュータとファイル転送するコマンドが、ftpコマンドです。このコマンドもtelnetコマンドと同様に、接続したコンピュータにユーザー登録をしていなければ接続したコンピュータとファイル転送を行うことはできませんが、ftpサーバーにあるファイルを受信する場合には、★ゲストでログインすることが許されています。

本学では、情報処理センターにあるsagaw (IPアドレス: 133.49.4.1) というコンピュータがftpサーバーです。

例えば、ogi (IPアドレス: 133.49.4.12) というホスト名を持つコンピュータとファイル転送をしたい場合は、次のように入力します。

★MS-DOSでは

```
A:¥> doskey
```

doskeyコマンドを実行することによりdoskeyコマンドの機能を利用することによりコマンド再実行が簡単に行うことができます。

```
A:¥> d F.8キーを押します
```

★MS-DOSでは

```
A:¥> F.7キーを押します
1 dir/w
2 date
3 time
A:¥> 1 F.9キーを押します
```

★telnetコマンド

telnetコマンドの実行でホスト名の入力を忘れた場合は、次の手順で接続します。

```
ユーザー名@himiko telnet
telnet: open
(to) ホスト名
```

また、telnetコマンドの終了時に、telnetプロンプトが表示された時は、まだtelnetコマンドを完全に終了していません。その場合は、次のように入力します。

```
telnet: quit
ユーザー名@himiko
```

★ゲストでログイン

ゲストのユーザー名として

anonymous

が用意されています。このユーザー名でログインするとパスワードは、メールアドレスを入力します。

```
ユーザー名@himiko ftp oqi
Name(ogi:ユーザー名): ユーザー名
331 Password required for ユーザー名
Password:パスワード
230 User ユーザー名logged in.
ftp>
```

接続したコンピュータのテキスト・ファイルを受信する場合は、次のように入力します。

```
ftp> ascii
ftp> get ファイル名
```

接続したコンピュータにバイナリ・ファイルを送信する場合は、次のように入力します。

```
ftp> binary
ftp> put ファイル名
```

ftpコマンドを終了させる場合は、次のように入力します。

```
ftp> bye
ユーザー名@himiko
```

本学のftpサーバー (sagagw) に接続する場合は、次のように入力します。

```
ユーザー名@himiko ftp sagagw
Name(ogi:ユーザー名): anonymous
331 Guest login ok, send your complete e-mail address as password
Password:メールアドレス
230 .....
ftp>
```

★ftpコマンド

ftpコマンドにはいくつかのコマンドが用意されていますので、主に使用するコマンドを紹介します。

open : コンピュータと接続する

close : 接続を終了する

bye : ftpコマンドを終了する

quit : ftpコマンドを終了する

ascii : 転送モードをASCIIにする

binary : 転送モードをバイナリにする

get : ファイルの受信

put : ファイルの送信

cd : 相手側のcdコマンドを実行する

ls : 相手側のlsコマンドを実行する

lcd : 自分側のcdコマンドを実行する

る

status : ftpコマンドの状態を表示

?: ヘルプ機能を実行

8 ファイルの編集について

UNIXにはファイルを編集するためのエディタがいくつか用意されています。テキストエディタとしてedとex、スクリーンエディタとしてviがUNIX標準エディタとして提供されています。また、フリーウェアとしてEmacsというスクリーンエディタがあります。

今回は、エディタとして一般的によく利用されているスクリーンエディタvi、Emacsの利用について紹介します。

8.1 viエディタの利用について

viには、

- [1] 各種機能を選択するための機能選択モード (コマンド・モード)
- [2] 書く機能にとまなう処理をするための処理モード
(テキスト・モード)
- [3] viの動作環境と終了条件などを設定するための拡張モード
(exモード)

の3つのモードがあります。

viを起動すると自動的にコマンド・モードとなり、次に機能を選択するためのコマンドを入力します。コマンドを入力するとテキスト・モードとなります。テキスト・モードからコマンド・モードへ戻るのには、基本的にはESCキーを押します。また、コマンドの代わりに” ; ” (セミコロン) を入力するとexモードになります。

(1) viの起動 [vi {ファイル名}]

viを起動するコマンドが、viコマンドです。

例えば、test.datというファイルをviで編集したい場合は、次のように入力します。

```
ユーザー名@himiko vi test.dat
```

test.datというファイルが新規ファイルであれば、次のような画面が表示されます。

```
~
~
~
" test.dat" [New file]
```

test.datというファイルが既存ファイルであれば、次のような画面が表示されます。

```
SAGA UNIVERSITY
Adachi Yumi
~
" test.dat" 3 line 28 bytes
```

(2) 新規ファイルの作成

新規ファイルを指定してviを起動させると画面左端に~ (チルダ) が表示されます。これは、空白行を意味しています。また、画面の下には、ファイル名とその状態 (新規ファイルだと、Newfile) が表示されます。

カーソルの位置は、先頭行の左端にありますので、文字の挿入モード (テキスト・モード) にするため、iを入力します。ここで、RETURNキーを押してはいけません。

文字の挿入モードになりましたので、自分の所属と氏名を入力してみましょう。また、文字の挿入モードは、ESCキーを押すまで有効ですので、文字の挿入ミスをしてそのまま最後まで入力し、入力が終了したらESCキーを押してコマンド・モードにしましょう。

文字の修正は、コマンド・モードにした後にカーソルを移動させテキストの削除と挿入の操作を繰り返して修正を行います。

★viの起動

vi ファイル名

★viの終了

ESCキーを押してコマンド・モードにします。

ZZ ファイルの編集結果を保存

: (セミコロン) を押すとexモードとなりカーソルが画面の下に移動します。

その後次のように入力します。

x ファイルの編集結果を保存

q! viを強制終了

コマンド・モードとテキスト・モードの切り換えは、ESCキーを押します。

コマンド・モードになっているとブレイク音があります。

★テキストの入力

i カーソルの左側に文字を入力

a カーソルの右側に文字を入力

I 現在行の先頭に文字を入力

A 現在行の末尾に文字を入力

o 現在行と次行の間に新しい行を挿入

O 現在行と前行の間に新しい行を挿入

J 現在行と次行を1行にまとめる

★テキストの削除

x カーソル位置の文字を削除

X カーソル位置の前の文字を削除

数xカーソル位置から行末方向に指定した文字数分を削除

数Xカーソル位置から行頭方向に指定した文字数分を削除

D カーソル位置から行末まで削除

dd 現在行を削除

★カーソルの移動

カーソル移動する場合は、コマンド・モードであることを確認してからカーソル移動コマンドを入力します。

↑or k 上方向

↓or j 下方向

←or h 左方向

→or l 右方向

★画面単位のカーソル移動

カーソル移動する場合は、コマンド・モードであることを確認してからカーソル移動コマンドを入力します。

CTRL-f 次画面へ移動

CTRL-b 前画面へ移動

CTRL-d 次半画面移動

CTRL-u 前半画面移動

```
i   SAGA UNIVERSITY
    Adachi Umi
    ~
ESC  ~
    " test.dat" [New file]
```

(2) ファイルの保存と vi の終了

ファイルの保存と vi を終了する場合は、次の手順で行います。

[1] ESC キーを押して、コマンド・モードにします。

ファイルの編集が終了したら、テキスト・モードからコマンド・モードに切り換えるため、ESC キーを押します。

[2] ZZ (SHIFT キーを押しながら Z キーを 2 回押す) を入力します。

編集したファイルを保存して vi を終了する場合は、コマンド・モードの状態 ZZ と入力します。

(3) 既存ファイルの編集

既存ファイルを編集するため vi を起動させると画面にファイルの内容が表示されます。また、画面の下には、ファイル名とその状態 (行数とバイト数) が表示されます。

```
SAGA UNIVERSITY
Adachi Yumi
~
~
" test.dat" 2 line 28 bytes
```

文字の修正は、カーソルを移動させテキストの削除と挿入の操作を繰り返して修正を行います。また、行の挿入や削除も文字の修正と同じように行いながら、ファイルの内容を編集していきます。

(4) 編集した内容を破棄して vi を終了

vi で編集した内容を破棄して vi を終了させる場合は、次のような手順で行います。

[1] ESC キーを押して、コマンド・モードにします。

ファイルの編集が終了したら、テキスト・モードからコマンド・モードに切り換えるため、ESC キーを押します。

[2] ; (セミコロン) を押して、ex・モードにします。

コマンド・モードから ex・モードに切り換えるため、; キーを押します。カーソルは画面下に移動します。

[3] q! を入力します。

編集したファイルの内容を破棄して vi を終了する場合は、ex・モードの状態 q! と入力します。

(5) vi の操作について

カーソルの移動やコマンド・モードとテキスト・モードの区別の表示がないなど、MS-DOS のエディタを使いなれているユーザーにとっては vi はたいへん不親切なエディタではないかと思えます。

また、ここで紹介しているコマンド群はよく使用するであろうコマンドだけを紹介しており、この他にもたくさんのコマンドが用意されています。

このように、ファイルの編集だけで UNIX アレルギーになってしまいそうですが、vi の操作は、何度もファイルを編集しながら操作をマスターしていきましょう。

★ vi 起動

vi ファイル名

★ vi の終了

ESC キーを押してコマンド・モードにします。

ZZ ファイルの編集結果を保存

; (セミコロン) を押すと ex モードとなりカーソルが画面の下に移動します。

その後次のように入力します。

x ファイルの編集結果を保存

q! vi を強制終了

コマンド・モードとテキスト・モードの切り換えは、ESC キーを押します。

コマンド・モードになっているとブレイク音になります。

★テキストの入力

i カーソルの左側に文字を入力

a カーソルの右側に文字を入力

I 現在行の先頭に文字を入力

A 現在行の末尾に文字を入力

o 現在行と次行の間に新しい行を挿入

O 現在行と前行の間に新しい行を挿入

J 現在行と次行を 1 行にまとめる

★テキストの削除

x カーソル位置の文字を削除

X カーソル位置の前の文字を削除

数 x カーソル位置から行末方向に指定した文字数分を削除

数 X カーソル位置から行頭方向に指定した文字数分を削除

D カーソル位置から行末まで削除

d d 現在行を削除

★カーソルの移動

カーソル移動する場合は、コマンド・モードであることを確認してからカーソル移動コマンドを入力します。

↑ or k 上方向

↓ or j 下方向

← or h 左方向

→ or l 右方向

★画面単位のカーソル移動

カーソル移動する場合は、コマンド・モードであることを確認してからカーソル移動コマンドを入力します。

CTRL-f 次画面へ移動

CTRL-b 前画面へ移動

CTRL-d 次半画面移動

CTRL-u 前半画面移動

8.2 Emacsエディタの利用について

Emacsは、UNIXが誕生する以前から使用されていた歴史のあるエディタで、たいへん奥が深く、その能力や機能のすべてをここでは紹介することは出来ません。そのため、ここではEmacsの基本的な操作方法だけを紹介いたします。

(1) Emacsの起動 [emacs {ファイル名}]

Emacsを起動するコマンドが、emacsコマンドです。

例えば、test.datというファイルをEmacsで編集したい場合は、次のように入力します。

```
ユーザー名@himiko emacs test.dat
```

test.datというファイルが新規ファイルであれば、次のような画面が表示されます。

```
[---]-***-NEmacs| test.dat          (EEE-|Fundamental)-All-----  
(New file)
```

test.datというファイルが既存ファイルであれば、次のような画面が表示されます。

```
SAGA UNIVERSITY  
Adachi Yumi
```

```
[---]-***-NEmacs| test.dat          (EEE-|Fundamental)-All-----
```

(2) 新規ファイルの作成と保存

新規ファイルを指定してEmacsを起動させます。Emacsの画面が表示されると画面上の左端にカーソルがあります。そこから、文字を入力していきます。文字の入力ミスした場合は、BSキーでカーソルを戻して再入力します。

```
SAGA UNIVERSITY  
Adachi Yumi
```

```
[---]-***-NEmacs| test.dat          (EEE-|Fundamental)-All-----
```

入力が終了したら、内容を保存してEmacsを終了する場合は、次の手順で行います。

[1] CTRLキーを押したままxキーとcキーを押します。

このキー操作で、Emacsを終了させます。

[2] ファイルの内容を保存するかしないかを聞いてきます。

ファイルの内容を保存して終了する場合は、yを入力します。

```
Save file /home/himiko/ユーザー名/test.dat?(y or n) y
```

ファイルの内容を保存しない場合は、nを入力します。そうするとこのままEmacsを終了するかしないかを聞いてきます。

```
1 modified buffer exists, do you really want to exit?(yes or no) yes
```

ファイルの内容を保存せずに終了する場合は、yesを入力します。もう一度、編集画面に戻りたい場合は、noを入力します。

★Emacsのキー操作

C-?

CTRLキーを押したまま文字を入力します。

M-?

ESCキーを押した後に文字を入力します。

★Emacsの起動

emacs ファイル名

★Emacsの終了

CTRLキーを押したままxキーを押した後cキーを押します。

C-x, C-c

★テキストの編集

C-SPACEキー マークをつける

C-@ マークをつける

C-w リージョンを削除し、キリング(バッファ)に保存

M-w リージョンをキリングに保存

C-y キリングの内容をカーソル位置に貼り付ける

★テキストの削除

DELキー カーソル位置の左側の文字を削除

C-d カーソル位置の文字を削除

C-k カーソル位置から行末まで文字を削除し、キリングに保存

C-w マーク位置からカーソル位置までを削除

★カーソルの移動

カーソル移動する場合は、コマンド・モードであることを確認してからカーソル移動コマンドを入力します。

↑orC-p 上方向

↓orC-n 下方向

←orC-l 右方向

→orC-b 左方向

C-a 行頭に移動

C-e 行末の移動

M-␣ ファイルの先頭に移動

M-␣ ファイルの末尾に移動

★ファイル操作のコマンド

C-x, C-f

ファイルの読み込み(編集可能)

C-x, C-r

ファイルの読み込む(編集不可能)

C-x, C-s

ファイルへ書き込み

C-x, C-w

ファイル名を変えて書き込む

★コマンドのキャンセル

C-G 実行中のコマンドをキャンセルする

(3) テキストの編集

Emacsでは、文字列の移動やコピーをカット&ペーストという方法で行います。この方法は、移動などをしてほしい文字列をカットして、それを移動させたい位置にペーストするものです。Macを使いなれたユーザーには、カット&ペーストの操作は、そんなに難しい操作ではないかと思えます。

Emacsでの文字列の移動やコピーは、一度バッファに入れておいて、その文字列を移動させたい位置にペースト（貼り付ける）することになります。

[1] 文字列の削除

削除したい文字列の先頭の文字にカーソルを移動させ、CTRL-SPACEキーを押すか、CTRL-@キーを押してマークをつけます。

次に、文字列の最後の文字（数行にまたがってもよい）にカーソルを移動させ、CTRL-wキーを押します。

[2] 文字列の移動

移動させたい文字列の先頭の文字にカーソルを移動させ、CTRL-SPACEキーを押すか、CTRL-@キーを押してマークをつけます。

次に、文字列の最後の文字（数行にまたがってもよい）にカーソルを移動させ、CTRL-wキーを押します。

次に、挿入したい位置にカーソルを移動させ、CTRL-yキーを押します。

[3] 文字列の複写

複写したい文字列の先頭の文字にカーソルを移動させ、CTRL-SPACEキーを押すか、CTRL-@キーを押してマークをつけます。

次に、文字列の最後の文字（数行にまたがってもよい）にカーソルを移動させ、CTRL-wキーを押します。

次に文字列を複写したい位置にカーソルを移動させ、CTRL-yキーを押します。

(4) Emacsのその他の機能について

Emacsには、ファイルの作成及び編集の他、メールの送受信やニュースの読み書きを行うことができます。

Emacs上で使用できるメールのプログラムのことをmh-e（エムエイチイー）といいます。このmh-eの操作は初心者にとっては、少々たいへんですので、Emacsの操作にある程度なれたところでチャレンジしてみましよう。

また、Emacs上で使用できるニュースのプログラムのことをGNUS（ニュース）といいます。このGNUSもmh-eと同じように初心者にとっては、少々たいへんですので、Emacsの操作にある程度なれたところでチャレンジしてみましよう。

★Emacsのキー操作

C-?

CTRLキーを押したまま文字を入力します。

M-?

ESCキーを押した後に文字を入力します。

★Emacsの起動

emacs ファイル名

★Emacsの終了

CTRLキーを押したままxキーを押した後cキーを押します。

C-x, C-c

★テキストの編集

C-SPACEキー マークをつける

C-@ マークをつける

C-w リージョンを削除し、キリング（バッファ）に保存

M-w リージョンをキリングに保存

C-y キリングの内容をカーソル位置に貼り付ける

★テキストの削除

DELキー カーソル位置の左側の文字を削除

C-d カーソル位置の文字を削除

C-k カーソル位置から行末まで文字を削除し、キリングに保存

C-w マーク位置からカーソル位置までを削除

★カーソルの移動

カーソル移動する場合は、コマンド・モードであることを確認してからカーソル移動コマンドを入力します。

↑orC-p 上方向

↓orC-n 下方向

←orC-f 右方向

→orC-b 左方向

C-a 行頭に移動

C-e 行末の移動

M-_ ファイルの先頭に移動

M-; ファイルの末尾に移動

★ファイル操作のコマンド

C-x, C-f

ファイルの読み込み（編集可能）

C-x, C-r

ファイルの読み込む（編集不可能）

C-x, C-s

ファイルへ書き込み

C-x, C-w

ファイル名を変えて書き込む

★コマンドのキャンセル

C-G 実行中のコマンドをキャンセルする

【付録.1】 Fortranの利用

情報処理センターのEWS (himikoを除く) でFortranを利用する場合は、下記のような手順で行います。

(1) Fortranを利用するための環境設定

Fortranを利用する場合は、最初にFortranを利用するための環境設定を行います。環境設定ファイル：.cshrcをvi,Emacs等のエディタで開き、pathの内容に/usr/lang/Fortran/ /home/lang/Fortran/を追加します。

```
set path=(. /usr/ucb /bin /usr/bin /usr/bin/X11 /usr/local/bin /usr/local/bin/emacs /usr/local/news /usr/local/bin/Wnn4 /usr/lang/Fortran/ /home/lang/Fortran/)
```

(2) Fortranプログラムのコンパイル

Fortranプログラムをコンパイルする場合は、下記のようにコマンドを入力します。

```
ユーザー名@ogi f77 Fortranプログラムファイル名
```

このコマンドを実行すると実行ファイル：a.outが生成されます。また、実行ファイルをtest.outというファイルに生成させたい場合は、次のようにコマンドを入力します。

```
ユーザー名@ogi f77 -o test.out Fortranプログラムファイル名
```

(3) 実行ファイルの実行

実行ファイルを実行する場合は、下記のようにコマンドを入力します。例えば、test.outという実行ファイルを実行したい場合は、下記のようにコマンドを入力します。

```
ユーザー名@ogi test.out
```

【付録.2】 TeXの利用

情報処理センターのEWSでTeXを利用する場合は、下記のような手順で行います。

(1) TeXを利用するための環境設定

TeXを利用する場合は、最初にTeXを利用するための環境設定を行います。環境設定ファイル：.cshrcをvi,Emacs等のエディタで開き、pathの内容に/usr/local/bin/tex /usr/local/bin/gnu/ghostscriptを追加します。

```
set path=(. /usr/ucb /bin /usr/bin /usr/bin/X11 /usr/local/bin /usr/local/bin/emacs /usr/local/news /usr/local/bin/Wnn4 /usr/local/bin/tex /usr/local/bin/gnu/ghostscript)
```

(2) TeXのコンパイル

TeXをコンパイルする場合は、下記のようにコマンドを入力します。

```
ユーザー名@himiko latex TeXソースファイル名
```

このコマンドを実行するとdviファイルが生成されます。

(3) dviファイルのプレビュー

dviファイルをプレビューする場合は、★X-Windowを起動させた後に、下記のようにコマンドを入力します。(dviファイル名には、拡張子は付けません。)

```
ユーザー名@himiko xdvi dviファイル名
```

このコマンドを実行するとディスプレイ上にTeXの文書がプレビューされます。

★X-Window

X-Windowは、センター内のEWS及びX端末でしか起動することができません。

学内LANに接続されているパソコンからは、TeXファイルのコンパイルとdviファイルのPSファイル変換、プリントアウトの実行ができます。

(4) dviファイルのプリントアウト

dviファイルをプリントアウトする場合は、下記のようにコマンドを入力します。(dviファイル名には、拡張子はありません。)

```
ユーザー名@himiko ldvi2kps dviファイル名 | lpr -PPS1
```

また、もう1台のプリンタに出力する場合は、次のようにコマンドを入力します。

```
ユーザー名@himiko ldvi2kps dviファイル名 | lpr -PPS2
```

(5) dviファイルをPSファイルに変換

dviファイルをPSファイルに変換する場合は、下記のようにコマンドを入力します。(dviファイル名には、拡張子はありませんが、PSファイル名には、拡張子:.psを付けます。)

```
ユーザー名@himiko ldvi2kps dviファイル名 > PSファイル名
```

このコマンドを実行するとPSファイルが生成されます。

(6) PSファイルのプレビュー

PSファイルをプレビューする場合は、X-Windowを起動させた後に、下記のようにコマンドを入力します。

```
ユーザー名@himiko ghostview PSファイル名
```

このコマンドを実行するとディスプレイ上にTeXの文書がプレビューされます。

【付録.3】 プリントアウト

情報処理センターには、2台のPS(ポストスクリプト)プリンタがネットワークに接続されており、EWSからプリントアウトを行うことができます。

(1) テキストファイルの出力

テキストファイルを出力する場合は、下記のようにコマンドを入力します。

例えば、プリンタ名:PS1のプリンタにテキストファイルを出力する場合は、下記のようにコマンドを入力します。

```
ユーザー名@himiko textout PS1 テキストファイル名
```

例えば、プリンタ名:PS2のプリンタにテキストファイルを出力する場合は、下記のようにコマンドを入力します。

```
ユーザー名@himiko textout PS2 テキストファイル名
```

(2) PSファイルの出力

PSファイルを出力する場合は、下記のようにコマンドを入力します。

例えば、プリンタ名:PS1のプリンタにPSファイルを出力する場合は、下記のようにコマンドを入力します。

```
ユーザー名@himiko lpr -PPS1 PSファイル名
```

例えば、プリンタ名:PS2のプリンタにPSファイルを出力する場合は、下記のようにコマンドを入力します。

```
ユーザー名@himiko lpr -PPS2 PSファイル名
```

9 参考文献

UNIX入門を執筆する上で、参考にした文献を紹介します。

- (1) たのしいUNIX -UNIXへの招待-
著者：坂本 文 発行：アスキー出版局 定価：1,900円
- (2) はじめてのUNIX
著者：佐々木 整 発行：秀和システムトレーニング(株) 定価：1,400円
- (3) 誰にでも使えるUNIX講座
著者：安岡 孝一 発行：ソフトバンク 定価：2,700円
- (4) すぐに使えるUNIX利用者から管理者まで
著者：松澤 茂・佐竹紀男 発行：共立出版 定価：2,900円
- (5) 入門NEmacs
著者：大木 敦雄 発行：アスキー出版局 定価：1,800円
- (6) ハッピー・ネットワーキング 新入生のためのインターネット入門
著者：山本 和彦 発行：アスキー出版局 定価：1,500円
- (7) MS-DOS5.0ユーザーズ リファレンスマニュアル