

シミュレーション概論ノート第1回(1999.10.20)

早川尚男

1 事始め

1.1 全般的注意

この授業で用いる計算機は Unix Workstation と呼ばれる計算機である。ワークステーションではパソコンとは異なって誰でも計算機を使用できる訳ではなく、スーパーユーザーと呼ばれる計算機を管理する人から認証された人だけがユーザー ID とパスワードを持っており、実際の計算機の使用を許可される。各種注意事項及び使用に関する注意書きは総合情報メディアセンター発行のセンターニュース No. 2,3 に詳しく説明されているのでよく読んでおくようにこのノートでは実際にプログラミングの時に最小不可欠の事柄について記す。本年度の講義内容については昨年度のノートのものをほぼ踏襲している。しかしながら本年度から半年の講義になったことや、前期の計算理学基礎論の履修を前提にするために若干の変更はある。

まず次の情報はしばしば必要になる。

- ここで配備されているワークステーションは HITACHI9000 VI132L-Tiny という機種である(どうでもいい情報である)。また外部から接続する際には `ws24.ip.media.kyoto-u.ac.jp` という IP address を持っていることである。この IP address を知っていれば Network に繋がっているマシンの何処からでも `telnet,rlogin` 等を用いて遠隔地からでも login することが可能になっている。
- 利用制限：ファイル容量 30MB¹ 及び、ファイルの個数 3000。
- ユーザー ID を忘れる人はいないと思うが password はしばしば忘れてしまうことがある。従って多少分かりやすいものに変更することを勧める。変更は `passwd`(或は `yppasswd`) とタイプせよ。そのときに古いパスワードを聞いてくるので入力、新しいパスワードも入力を求められる。再入力して変更完了。しかしなが意味のある単語等

¹ bdf というコマンドで確認できる。

から成る password は security 上の問題をもたらすので数字や大文字を混ぜたり、定期的な変更をしたりする注意が必要である。²

以上、箇条書にしたことはいずれも重要な注意である。しばしば必要になるので常に頭においておくこと。

1.2 login,logout

利用の開始についてはセンターニュースに詳しい説明がある。必要なのは電源を入れて、login と聞かれたら user ID をいれて Enter キーを押す。次にパスワードを入力して Enter キーを押すと login に成功する。そうすると大きなウインドウ (四角いフレームで囲まれたもの) が現れるのでそこで計算機の操作を行なうことができる。初期設定では、ウインドウマネージャには fvwm2.0.4 が使われるようになっている (1998/02/01 現在)。キーボードからの入力が有効なウインドウは、縁が若草色になっています。Window95 と違って、必ずしも一番手前のウインドウがいつも有効というわけではないことに注意。

あるウインドウを一番前にしたい場合には選びたいウインドウのタイトルバーにマウスカーソルを合わせて、マウスの左 ボタンクリックすればいい。

またルートメニューと呼ばれるものが存在する。マウスを移動させて背景部分に持っていき、左ボタンを押すとメニューが現れる。

利用の終了にはルートウインドウ (背景の黒い画面) でマウスの左ボタンをクリックする。メニューが表示されるので一番下の「Quit...」を選択する。了解を求めて来るので中央の「Yes, Really Quit」を左ボタンでクリックすれば login の画面に戻る。

引続き利用せず電源を落す場合には利用者名称 (Login) のところに shutdown と入力すること。Password の部分には何も入力しないで Enter key を押すこと。1 分程でうまく shutdown が出来たメッセージ (System has halted.) が表示されるので、確認後、電源ボタンを押して電源を切断して下さい。

² システム全体でパスワードの変更が有効になるのにはしばらく時間がかかり、場合によっては 1 時間程かかるので注意が必要である。

1.3 First Step of Unix

Unixのコマンドについては利用の手引 Chapter 4 に解説してある。特に

- cat 引数に指定されたファイルの内容を表示
- cd カレントディレクトリーの移動
- chmod ファイルやディレクトリーの保護モードの変更
- cp ファイルのコピー
- ls ディレクトリーの下にあるファイルのリストの表示
- mkdir ディレクトリーの作成
- more 指定ファイルの内容を画面に合わせて1ページずつ表示
- mv ファイル名の変更
- pwd カレントディレクトリーの表示
- rm ファイルの削除
- rmdir ディレクトリーの削除
- tr 文字の変換

は代表的なコマンドである。詳細は適当な Unix の参考書か、昨年まであった「利用の手引」を参照せよ。

利用の手引を忘れた場合や他のコマンドの詳細を知りたい場合はオンラインマニュアルを使うと便利である。例えば

```
% man chmod ↵
```

と入力すると(↵はリターンキーを押す操作)、chmod コマンドの詳細が表示される。

1.4 Mule 入門

計算機上でプログラムを作成する場合や文書を作成する場合に mjpgterm 上で書き連ねても文書ファイルを作成しない。あくまであるファイルの中身を作成するのは編集をするためのエディタを用いて打ち込む作業が必要になる。

Unixには様々なエディタがあるがここではその中でも最もポピュラーで高度な機能を備えたエディタとして Mule がある。その詳細はセンターニュース No.3 に譲るとして、極く簡単に必要不可欠なことだけを紹介する。

Mule を起動するにはルートウィンドウから左ボタンで Root Menu を選択し、その中の Mule を選択する場合と kterm 等で

```
% mule & ←
```

と入力する方法がある。ここで & を加えたのは Mule 使用中も kterm を使用可能な状態に保つためである。またファイル名を指定して

```
% mule filename & ←
```

とした場合にはそのファイルが存在する場合は該当ファイルを呼び出すし、また存在しない場合は新しいファイルの作成の操作に入ったとして認識される。

Mule 上で必要なコマンドの殆んどはエコー領域にあるメニュー画面を覗くことで自動的に実行できる。それらのコマンドは勿論、メニューを経由しなくても実行できる。代表的なコマンドは

- C-x C-c : Mule の終了
- C-x C-x: 編集中ファイルの保存
- C-\ : Wnn6 の起動 (日本語が使えるようになる) と終了
- C-x C-w ファイル名を変更して保存
- M-x recover-file 自動セーブファイルから元のファイルを復元
- M-x function: function の実行
- C-x m: mh(電子メールシステムの起動)

きりがないのでやめます。センターニュース No.3 から簡単な使い方を講習する。但しこれらのコマンドで C と記してあるのは CTRL キーを表し、M と記してあるのは ESC キーを表している。

1.5 日本語入力方法

日本語入力は、まずローマ字かな変換モード (日本語モード) にしてから、ローマ字かな変換でひらがなを入力したあと、漢字変換をして漢字に直す。

1.5.1 日本語モードへの切り替え

C-\ (Ctrl キーを押しながら \ キーを押すという意味) でモードを切り替える。日本語モードになった証拠に、Mule ウィンドウの下から 2 行目の反転文字行 (モード行といえます) の左のあたりに、[あ] と表示される。この状態でひらがな (だけ) 入力できる。ちなみに、日本語モードから英数モードに戻るときは、未変換文字列がない状態で C-\ を行なう。

1.5.2 ひらがなの入力

日本語モードにおけるかな変換方式は、ローマ字かな変換方式です。日本語モードになっていることを確認してから `kyouhaiitenkidesune.` と入力すると、| きょうはいいてんきですね | となる。

1.5.3 ひらがな漢字変換：その 1

ひらがな漢字変換の開始は、この状態で、SPACE キーを押すことによって始まります。

初めてこれを行う場合、Mule の最下行に「頻度 ファイル “/home/.../.Wnm6/fisd.h” がありません。作りますか? (y or n)」と聞かれるので、y キーを押す。このようなメッセージがあと連続して 6 回出るので、全部で 7 回 y キーを押すこと。

この例では、「きょうは」を「今日は」と漢字にしたいところです。カーソルは漢字候補を入れ替えたい文頭の文節にあるので、そのまま「きょうは」に相当する文節について、カーソルキーの `h` キーで次々に候補を入れ替えられます。候補が確定して全ての文節が正しい漢字表現になったら、Enter キーを押すことで確定させます。

1.5.4 ひらがな漢字変換：その 2

入力されたひらがなに対して、SPACE キーで漢字変換を開始したとき、文節の切れる位置がおかしいときがあります。このようなときは、カーソルキーで切目がおかしい部分の左側の文節に文節を移動させ、以下の操作をします。

Ctrl キーを押しながら i キーを押すことで、文節が 1 文字分伸びる

Ctrl キーを押しながら o キーを押すことで、文節が 1 文字分縮む

2 メール

課題を出す場合、その他でメールを使う必要があるかもしれない。その場合は C-x m で mh を立ちあげた後、s と入力すると To: という画面が現れるのでそこで相手のアドレス (例えば hayakawa) を指定する。更に Subject その他を入力するとメールの編集画面になる。そこで内容を入力した後、C-c C-c でメールを送信できる。他にも様々なメーラーがあるので各自の好みに合わせて用いれば良い。

総合情報メディアセンターでは Mew がメーラーとして標準装備される様になっている。以下で極く簡単にその使い方を紹介しよう。

2.1 Mule 上の Mew をメールリーダーとして使う

2.1.1 最初るとき

kterm の中で、inc コマンドを実行します。質問には y キーを押して、Enter キーを押して下さい。

```
us201:~% inc
Your MH-directory "/home0/fcl/y/ykameda/Mail" doesn't exist;
Create it? y
inc: no mail to incorporate
us201:~%
```

2.Mule を起動します。

3.M-x mew (Esc キーを押したあと、x キーを押し、さらに mew と入力すると、このように Mule の一番下の行に表示されます。その後で、Enter キーを押します。)します。

4.Mule 最下行に以下のような表示が出ますので、Enter キーを押します。

Range (update):

すると、メールの読み書きが可能な状態になります。

ここまで来たところで、幾つかの使い方があります。

2.2 Mew でメールを出す

w キーを押すと、新規メール作成状態になります。

```
To:
Subject:
X-Mailer: Mew version 1.92 on Emacs 19.34 / Mule 2.3 (SUETSUMUHANA)
Mime-Version: 1.0
-----
```

と Mule 内に表示されますので、To: に続いて相手のメールアドレスを入力して下さい。Subject: の続きには題目を簡潔に書きましょう。また、本文は必ず “-” が 4 つ続いている行の下から書きはじめて下さい。

ここは練習なので、自分に送ってみましょう。USERID@ip.media.kyoto-u.ac.jp あるいは USERID を To:以下に書いてみて下さい。書き終えたら、メールの送出行うために C-c C-c (Ctrl キーを押しながら c キーを押す、さらに もう一度 Ctrl キーを押しながら c キーを押す操作) をします。

すると、下のようなメッセージが Mule ウィンドウ最下行にでます。

```
Content-Type: was automatically added. Send this message? (y=RET
or n=SPC)
```

ここで、Enter キー (ないし y キー) で実際に送られ、SPACE キー (ないし n キー) だとまたメール文章作成中の状態に戻ります。

2.3 Mew でメールを読む

新規にメールが到着しているかどうかは、Mew の起動状態で、i キーを押して確認できます。i キーを実際に押すと、図 109 のように問い合わせをしてくるので、Enter キーを押して下さい。

ここでの例では、先ほど自分に送ったメールが到着しているのことで確認して下さい。

メールが来ていた場合は、メール一通につき一行の表示で一覧が Mule ウィンドウ内に出ます。n キー、p キー でカーソルを上下させると、その行に応じたメールが下半分に表示されます。ないしは、読みたいメールの行へ キーでカーソルを上下させて合わせ、Enter キーないし SPACE キーを押しても同様に表示されます。

2.4 Mew でメールに返事する

メールに返事をしたい場合には、n キーや k キー、`^` キーで返事したいメール行にカーソルを合わせます。そこで、a キーを押すと、返事メール作成バッファが出来ますので、あとは同様にメールを送信します。

3 WWW

本講義では講義ノート WWW 上に公開し、それをダウンロードするというスタイルで行なう。従って WWW の利用は大事になる。

日本語版 Netscape Communicator 4.04 が用意されています。起動法その他についての詳細はスタートアップガイド <http://www.imel1.kuis.kyoto-u.ac.jp/members/kameda/lecture/cims/startup/startup203html/startup203html.html> にある WWW の項を参照して下さい。

注意：以降の説明は、fvwm のメニューで起動できる netscape に対してではありません。下に述べるように、kterm から起動してください。

3.1 起動方法

kterm の中で `jnetscape` と入力して起動します。

3.2 はじめてのとき

Netscape 社との契約及び初期化ウィンドウあなたが総合情報メディアセンターのワークステーションで初めて Communicator4.04 を起動するならば、4 つほどのウィンドウが出ますが、その場合はいずれも OK をクリックして下さい。

これらのウィンドウは 2 回目以降でくることはありません。最終的に、Netscape Communicator4.04 が起動されます。

総合情報メディアセンターの Web は既に紹介した通り <http://www.media.kyoto-u.ac.jp/> です。そこで今日紹介した様々な情報が詳しく説明されています。

4 プログラミング言語

ようやく準備が終ってプログラミングに入ることが出来る。しかしながらここでいきなり、本講義の目的である数値計算に入るにはまだ早い。実際にはある種のプログラミング言語の初歩的なルールを知らないといけない。標準的な言語として Fortran と C が挙げられる。特に Unix ツールの殆んどは C 言語によって書かれているので今や Fortran を学ぶ価値は殆んどなく C を知っていれば充分と言ってよい。しかしながら Fortran をこの授業での標準言語として用いるのは個人的な理由³ の他に、前期に計算理学基礎論によって Fortran90 の講義が用意されていることが大きい。また Fortran が単純な繰り返し計算等大規模な数値計算に適していること、歴史的に数値計算は Fortran で行なわれてきたためにライブラリその他の蓄積があることにもよっている。とりあえず本講義ノートでは可能な限り両方の言語を併記しておきたい。

またこのコンピューターはパスが通っていないのでプログラムの実行等が不便である。従って最初に Home directory に .cshrc という file をつくって

```
set path=(. $path)
```

と書き save しよう。この file が出来ているかどうかは `ls -a` で確認できる。

また総合情報メディアセンターでは Fortran として Fortran90 を用いている。Fortran90 は Fortran77 を包含する形で拡張しており、更に C で用いられていた様々な用法を採り入れてその差は従来に比べると減少している。現在まで Fortran90 の文法書はあまり出版されておらず、一般の人たちが Fortran77 との違いを必ずしも認識していない。その点については富田博之教授による計算理学基礎論の本に詳しく載っている。

注意しておかなければならない点はプログラミングにおけるアルゴリズムそのものは言語の詳細によらない。むしろ、良く整備されたプログラムでも他人の作成したものを読みこなすのは非常に難しい。私が高橋大輔著「数値計算」(岩波)を参考書として指定した最大の理由は従ってこの本がプログラム例を載せずにアルゴリズムを明解に示しているからである。その意味で言語の選択はそれほど重要な問題ではない。

³ 勿論、私が昔、Fortran77 を学び、未だに Fortran User であることが最大の原因である

プログラムを書く際に大切な点は分かりやすさである。一見して読むのを放棄したくなるようなプログラムは大概間違いを含んでいるし、作った本人も時間が経って読み返してみても判読不能である。

5 Fortran90の注意事項

ここで Unix での Fortran の実行についてごく簡単にまとめておこう。

5.1 識別子 (拡張子)

この二つのコーディング形式はコンパイル時にオプション指定して区別するが、ソースプログラムのファイル識別子を以下のようにすることでデフォルト解釈される：

固定形式プログラム： `.f` (注：パソコンでは `.for`)

自由形式プログラム： `.f90`

またこの講義ノートでは全て Fortran77 に準拠した固定形式で書かれている。

例題 1： $\sum_{i=1}^{10} i$ を求めよ。

Fortran77 或は Fortran90 固定形式で書けば

```
integer x,i

x=0

do i=1,10
  x=x+i
enddo

write(*,*) 'x=',x
end
```

という風になる。これを Fortran90 自由形式で書けば

```
integer x,i
x=0
do i=1,10
```

```

x=x+i
enddo
write(*,*) 'x=',x
end

```

とできる。違いは拡張子の他に左詰めができたかどうかである。Fortran77及び固定形式では左端の6文字分は行番号や&記号に当てられるために、そこには文字列を入力できないことは注意しなければならない。

Mule 上に上の様なプログラムを書いたとしてもそれだけでは答えは得られない。そのためには次の手順が必要である。

(i) エディタを用いてプログラムを作成する。

(ii) そのプログラムをファイルとして格納する。その時拡張子 *.f* が必要である。例えば上のプログラムを *sum.f* (自由形式であれば *sum.f90*) というプログラムとしてセーブするとしよう。

(iii) プログラムをコンパイルする。一番簡単には

```
f90 sum.f ←
```

とする。こうすると *a.out** というファイルが出来ることは容易に確認できる⁴。次に

```
a.out ←
```

と⁵行なうと $x = 55$ と画面上に表示される。また

```
a.out>a.dat ←
```

とすると *a.dat* というファイルが出来てそこに1行、 $x = 55$ と書いてある。

6 C言語によるプログラムの基本構造

ここの章は必ずしも内容がまとまっていない。それは私がCに習熟していないからである。そのせいもあってここではスタイルを変えて帰納的に議論を進めてFortran90との比較を試みよう。

6.1 例題

簡単な例題として次の計算を試みよう。

例題1: $\sum_{i=1}^{10} i$ を求めよ。

⁴ ls を使うべし

⁵ パスが通っていない状態では *./a.out* とする。

計算しなくても答えは 55 であることはわかる。これをアルゴリズムとして書くと

- (1) $x := 0$ とする。
- (2) $i := 1, 2, \dots, 10$ の順に

$$x := x + i$$

を繰り返す

- (3) x を答えとして表示する。
ということになる。

```
/* program for summation */
#include <stdio.h>
#include <math.h>

main()
{
  int i,j;

  j=0;
  for (i=1; i<=10; i++) {
    j=i+j;
  }

  printf("sum=%3d\n",j);
}
```

となる。最初の行の

```
/* program for summation */
```

は注釈である。Fortran90 と異なり注釈は必ず

```
/* */
```

でくくられなければならない。

次の

```
#include <stdio.h>
#include <math.h>
```

は標準ライブラリと呼ばれ、順に入出力の関数などに必要な定数、sin, cos などの数値関数の定義ファイルの読み込みを指示している。stdio は STandarD Input Output の略である。他にも様々なライブラリがあるが我々にとってこの2つが最も重要である。他については随時紹介しよう。

6

C 言語では main () という一番おおもとの関数が必要であり、

```
main () {
    本体
}
```

のように書かれる。本体では最初に変数や定数の宣言が行なわれた後に本文となる。変数は必ず定義されなければならない点に注意が必要である。Fortran の Write 文に対応するのが printf である。printf の " に囲まれた文字列は書式の指定で、%3d はその位置に 3 文字分の整数を表示する (実数は %f とか %8.4f などとなる)。また \n は改行を意味する。

注意すべきは大文字と小文字の区別がある点である。

また各行 (文) の終りに必ずセミコロン ; をつけることを忘れてはならない。逆に言えばセミコロンなしでは一つの命令が完結しない。

6.2 プログラムの実行

C のプログラムの拡張子は .c である。例えば前述のプログラムを sum.c に save したとすると、コンパイルは

```
gcc sum.c -lm
```

で行なう。l は library の略で libmath という数学用のライブラリを読み込むことを意味している。実行は Fortran と同様に

```
a.out
```

で行なえばよい⁷。

⁶ 詳しくは C 言語の文法書を読むこと。

⁷ パスが通っていない場合は ./a.out 。

6.3 データの形

Cで使える基本変数の形は

char	1文字または1バイトのデータ
int	整数
float	単精度の浮動小数
double	倍精度の浮動小数

である。

6.4 プログラムで扱える構造

(2つの数値データの間的大小関係式)

大きい	>
大きいか等しい	>=
小さい	<
小さいか等しい	<=
等しい	==
等しくない	!=

論理演算子

論理否定	!
論理積	&&
論理和	
論理等価	==
論理非等価	!=

条件判断

```
if (a>b) {  
    . . . . ;  
}else if (a==b) {
```

```
    .... ;  
} else {  
    .... ;  
}
```

単純な繰り返し

```
for (i=1; i<=10; i++) {  
    statement 1  
    statement 2  
}
```

などである。

組み込み関数などは別紙参照。