卒研でのコンピューター (2) 改訂版

桂田 祐史

2004 年 4 月 27 日

1 連絡事項

1. Windows と Solaris で日本語の文字コードが異なっています。対処法は色々ありますが、なるべく次のようにして下さい。DELL のマシン (ashigaru*) の kterm から Solaris にログインすると、入出力の日本語の文字コードを sjis にするようになっているので、Windows での kterm を sjis モードで使ってください。設定ファイルを更新すると良いでしょう。

.xinitrc の更新

```
ASHIGARU02% cp -p y:/local/skel/.xinitrc z:/.Windows↵
```

まとめておくと

	日本語文字コード	行末	変換法
Windows	シフトジス	CR, LF (C-m, C-j)	`qkc -ms` ファイル名
Solaris	日本語 EUC	LF (C-j)	`qkc -eu` ファイル名

2. DELL パソコンの Windows 環境でテキスト・ファイルの編集をするには、**meadow** を使うことをお奨めします (コマンド `mule` でも起動できます)。これは GNU emacs の Windows への移植で、日本語がきちんと使えます。

2 前回の復習

2.1 数学科のコンピューター環境

『数学科でコンピューターを使う』という文書を見よ。これは『桂田研 2004 年度卒研』[1] から読むことが出来る。

2.2 パソコンの利用手順

- お奨めは DELL のパソコンを使うこと。

[1] `http://www.math.meiji.ac.jp/~mk/labo/2004/`

- 電源は自分で ON にしてよい。MATH6701 ドメインにログオン。
- Mathematica や MATLAB は Windows 版で十分である (こちらの方が使いやすい?)。
- C や C++ でプログラミングするには、現時点では UNIX 環境 (Solaris WS) が便利と思われる。X (X Window System) を起動してから、UNIX にログインするのがよい。

UNIX へのログイン

1. デスクトップにある Cygwin Tcsh というアイコンをクリックし、`startx` を実行する。久しぶりの場合、設定ファイルを次のようにして変更するとよい。

こうやって更新

```
ASHIGARU02% cp -p y:/local/skel/.* z:/.Windows↵
```

2. kterm で `slogin oyabun` として oyabun にログインする。終了したら `logout` コマンドでログアウトするのが良い。

さあ終了だ

いきなり Windows をログオフしても大抵の場合は大丈夫ですが、「自分で掃除する」には次のような手順を踏みます。

1. emacs などはきちんと終了するのが望ましい。
2. Solaris からも logout するのが望ましい。
3. タスクバーで右クリックして、X を終了する。
4. Cygwin Tcsh で exit コマンドを実行して、ウィンドウ右上のボタンをクリックして閉じる。

2.3 C によるプログラミング (整理)

GCC (GNU Compiler Collection) の C コンパイラーである **gcc** の利用を勧める[2]。
`gcc` コマンドだけでなく、 `ccv`, `ccmg` などのコマンドも gcc を使っている[3]。
C プログラムを入力・編集するために `emacs` が使える。

`prog.c` を emacs で編集する

```
emacs prog.c &
```

`prog.c` をコンパイルして実行可能プログラムを作るには、例えば gcc を用いる。

[2]GCC には、C++ コンパイラー g++, Fortran 77 コンパイラー **g77** もある。
[3]何をしているか知りたければ、`which ccv` とか `cat /usr/local/meiji/bin/ccmg` としてみよう。

とにかくコンパイル

```
gcc prog.c
```

とすると a.out という実行可能プログラムができる。実行するには

a.out を実行する

```
./a.out
```

とする[4]。

実行可能プログラムの名前を指定することも出来る。

-o で実行可能プログラムの名前を指定する

```
gcc -o prog prog.c
./prog
```

数学関数ライブラリィをリンクするには -lm とする (l は L の小文字)。また最適化オプション -O も指定した方が良いかもしれない (O は Optimize の 'O')。

最適化、実行可能プログラム名の指定、数学関数ライブラリィのリンク

```
gcc -O -o prog prog.c -lm
```

数学科標準の設定では、cco というエイリアスを定義してある。

簡単なものは cco あるいは ccv で OK

```
cco prog.c
./prog
```

とすればよい。

3 C のプログラムでグラフを描く

3.1 C のグラフィックス事情 — GLSC のススメ

規格に則った (グラフィックスを使わない[5]) プログラムならば、UNIX でも Windows でも共通のソース・プログラムですむが、グラフィックスを用いるプログラムは環境依存になる。明治大学数学科では **VOGLE**[6](数学と計算機・演習の ccv コマンド), **GLSC** などのグラフィックス・ライブラリィが使われている。

卒研では GLSC の利用を推奨する。

GLSC については、http://www.math.meiji.ac.jp/~mk/labo/howto/index.html#GLSC にリンクを張ってあるが、特に

『明治大学数学科計算機室ユーザーのための GLSC の紹介』[8]

[4]情報科学センターの実習では、./ を省略して、単に a.out と出来たかもしれない。これは path にカレント・ディレクトリィ '.' が含まれていたためである。最近はトロイの木馬を防ぐために、カレント・ディレクトリィは path に含めないようになっている。

[5]先日渡した本 (柴田著) にはグラフィックスはない。

[6]VOGLE に使い方については、『VOGLE グラフィックライブラリの紹介』[7] を見よ。

[8]http://www.math.meiji.ac.jp/~mk/labo/howto/intro-glsc/

という文書を用意してある。

3.2 練習

少し先の目標はテキスト 7 章『発展系の数値解析』の数値実験であるが、まずは常微分方程式の初期値問題を解いて、可視化することを目標にする。

古いですが `http://www.math.meiji.ac.jp/~mk/soyri2-1995/` にある文書が参考になるかも。

- 第 3 回『グラフの描き方 (2)』
- 第 6,7,8 回『常微分方程式の初期値問題の数値解法』

1. 自分が選んだ 1 変数関数のグラフを描くプログラムを作り、印刷せよ (関数を替えたときに変更がすぐに出来るように工夫すること)。
   こういうのは自分で選んで色々試すのが良いが、最初は何か指示してくれと言うのならば

$$f(x) = \begin{cases} x \sin \dfrac{1}{x} & (x \neq 0) \\ 0 & (x = 0) \end{cases}$$

   でいかが？ (場合わけがあるので難しいかな?)

2. 自分が選んだパラメーター曲線を描くプログラムを作れ。
   (例えばアルキメデスの螺旋 $r = \theta$ $(\theta \geq 0)$)

3. 常微分方程式の初期値問題

$$x'(t) = x(t), \quad x(0) = 1$$

   を数値計算で解け。解曲線を描け。

4. 常微分方程式の初期値問題

$$x''(t) + px'(t) + qx(t) = 0, \quad x(0) = A, \quad x'(0) = B \quad (p,\, q,\, A,\, B \text{ は実定数})$$

   を数値計算で解くプログラムを書け。解曲線 (横軸 t, 縦軸 x のグラフ) を描け。

5. 常微分方程式の初期値問題

$$\begin{aligned} x'(t) &= ax(t) + by(t) \\ y'(t) &= cx(t) + dy(t) \end{aligned}$$

$$x(0) = x_0, \quad y(0) = y_0$$

   を数値計算で解き、解曲線を描く (t をパラメーターと見て、$(x(t), y(t))$ の軌跡を描く) プログラムを作れ。