

# 応用数値解析演習 Cygwin 実行手順マニュアル

応用数値解析演習を行うにあたり、サンプルプログラムのコピー、Cygwin, X-window の起動、そしてサンプルプログラムの実行までを簡単に説明する。

## 1 Cygwin を起動する前の準備

下記の手順で Cygwin 用の home ディレクトリを作成し、サンプルプログラムを共有ディスクからコピーしてくる。

- ( 1 ) 共有ディスク 「E: ¥ 」を開く
- ( 2 ) 「E: ¥ mech\_ana2007 」を開く
- ( 3 ) 「start.cmd」 をダブルクリックする

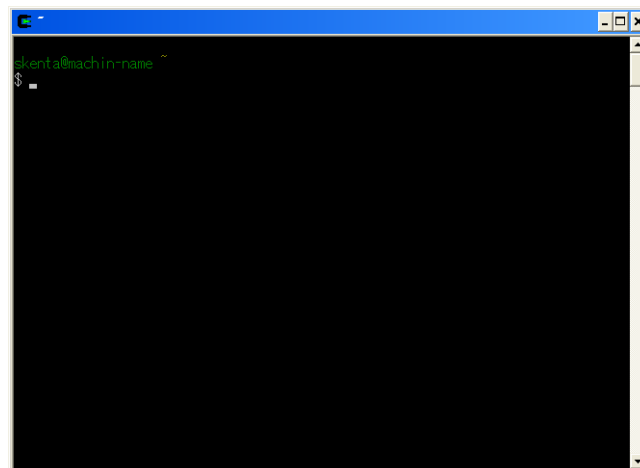
バッチスクリプト「start.cmd」を実行（ダブルクリック）すると、`C: ¥ cygwin ¥ home ¥ guestXX ¥` が作成され、その中に `¥ samples` というフォルダがあり、1D-sample, 2D-sample というフォルダが入っている。

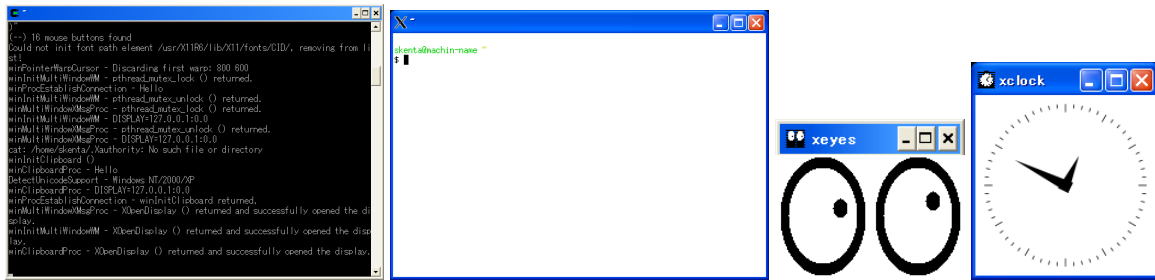
`C: ¥ cygwin ¥ home ¥ guestXX ¥ samples ¥ 1D-sample`

という階層構造になっている。「guestXX」は各マシンにログインしたユーザ名になっている。

## 2 Cygwin の起動と X-window の立ち上げ

Windows 画面の Cygwin アイコン、もしくは 「すべてのプログラム」 「Cygwin」 「Cygwin Bash Shell」から Cygwin を起動すると以下のような黒いターミナルが表示される。  
この中では、ls や cd といった Unix コマンドを実行することができる。





サンプルプログラムの多くは X-window 上で実行する必要がある。

Cygwin のターミナルで `$ startx` と打ち込むとターミナル上に数十行にわたるテキストが表示され、「xterm」という X-window の (白い) ターミナルが立ち上がる。

xterm で `$ xeyes` や、 `$ xclock` といったコマンドを入力すると X-window のアプリケーションが動作する。  
`$ xeyes &` のように各コマンドの後にスペースを開けて「&」を入力するとコマンドがバックグラウンドで実行される。

### 3 サンプルプログラムの実行

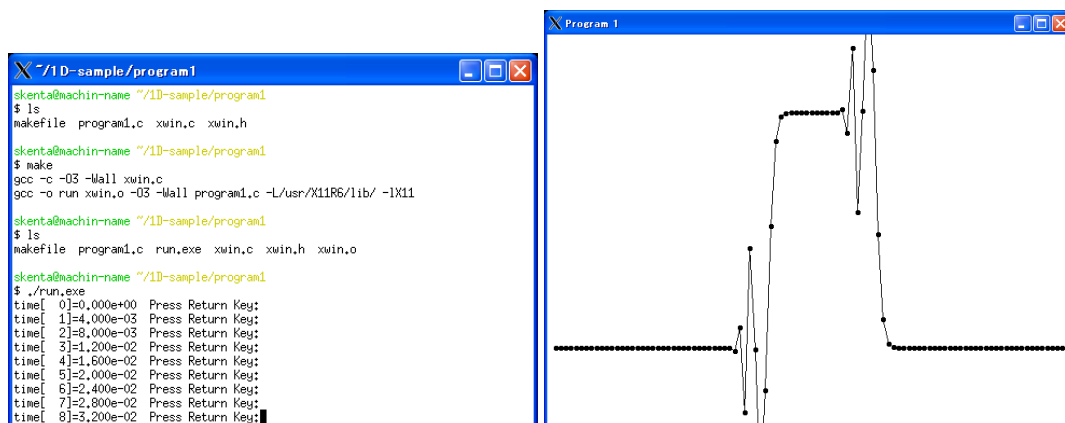
サンプルプログラム (program1) を実行する。まず、xterm で

```
$ cd samples/1D-sample/program1
```

と打ち込んで「program1」というフォルダに移動する。そこには「makefile」、「program1.c」、「xwin.c」、「xwin.h」という4つのファイルがあるのが `$ ls` と打ち込んで確認できる。

- ・「makefile」 コンパイル等の情報が書いてある
- ・「program1.c」 サンプルコード
- ・「xwin.c」 X-window にグラフを表示する関数が書いてある
- ・「xwin.h」 xwin.c のヘッダーファイル

`$ make` と入力することで「run.exe」、「xwin.o」というファイルが生成される。`$ ./run` と打って実行すると (「`./run.exe`」でもよい) ターミナルに「Press Return Key:」と表示され、Return キーを押すたびに計算が進むようになっている。



## Appendix

### Unix ターミナルコマンド

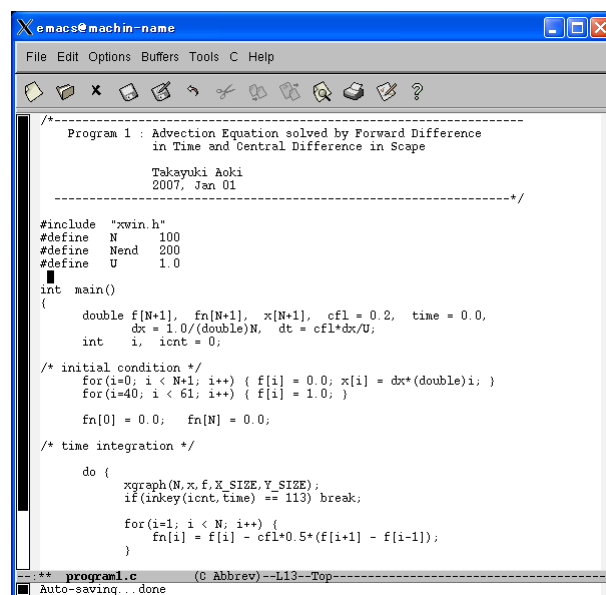
| 基本的なターミナルコマンド  | 説明  |
|--|---|
| ls   | ファイルを表示する   |
| pwd  | 現在いるディレクトリを表示する   |
| cd   | home ディレクトリに移動する  |
| cd <span style="border: 1px solid black; padding: 0 5px;">フォルダ名</span>   | <span style="border: 1px solid black; padding: 0 5px;">フォルダ名</span> に移動する |
| cd ../   | 1 つ下位のフォルダに移動   |
| cd -   | 1 つ前の作業フォルダに移動  |
| cp <span style="border: 1px solid black; padding: 0 5px;">コピー元名</span> <span style="border: 1px solid black; padding: 0 5px;">コピー先名</span>         | ファイルをコピーする  |
| cp -r <span style="border: 1px solid black; padding: 0 5px;">コピー元名</span> <span style="border: 1px solid black; padding: 0 5px;">コピー先名</span>      | フォルダごとコピーする   |
| display <span style="border: 1px solid black; padding: 0 5px;">画像ファイル</span>   | 画像ファイルの表示   |
| animate <span style="border: 1px solid black; padding: 0 5px;">連番ファイル</span>   | 連番画像ファイルのアニメーション表示  |
| convert <span style="border: 1px solid black; padding: 0 5px;">元ファイル</span> <span style="border: 1px solid black; padding: 0 5px;">変換後のファイル</span> | 画像フォーマットの変換   |
| import <span style="border: 1px solid black; padding: 0 5px;">出力ファイル名</span>   | X-window 画面を各種画像ファイルにダンプ  |

### テキストエディタ

プログラムを編集するには vi や emacs といったテキストエディタがよく使われる (メモ帳でもよい)。vi は高機能なエディタであるが初心者にはとっつきにくいので emacs を紹介する。

\$ emacs program1.c &

と打ち込むと emacs で program1.c を開くことができる。編集した内容を保存するには「File」  
「Save(current buffer)」, あるいは Control-x Control-s とすればよい。



```
emacs@machin-name
File Edit Options Buffers Tools C Help

/*-----
Program 1 : Advection Equation solved by Forward Difference
in Time and Central Difference in Space

Takayuki Aoki
2007, Jan 01
-----*/

#include "xwin.h"
#define N 100
#define Nend 200
#define U 1.0
int main()
{
    double f[N+1], fn[N+1], x[N+1], cfl = 0.2, time = 0.0,
    dx = 1.0/(double)N, dt = cfl*dx/U;
    int i, icnt = 0;

    /* initial condition */
    for(i=0; i < N+1; i++) { f[i] = 0.0; x[i] = dx*(double)i; }
    for(i=40; i < 61; i++) { f[i] = 1.0; }

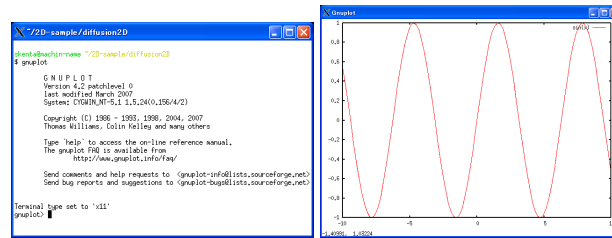
    fn[0] = 0.0; fn[N] = 0.0;

    /* time integration */
    do {
        xgraph(N, x, f, X_SIZE, Y_SIZE);
        if(inkey(icnt, time) != 113) break;
        for(i=1; i < N; i++) {
            fn[i] = f[i] - cfl*0.5*(f[i+1] - f[i-1]);
        }
    } while(1);

    return 0;
}

--** program1.c (C Abbrev)--L13--Top--
Auto-saving... done
```

## gnuplot



X-window で使えるグラフ作成ツールに「gnuplot」がある。xterm で

```
$ gnuplot
```

と打つと起動し、

```
$ q
```

で終了する。

```
gnuplot > plot sin(x)
```

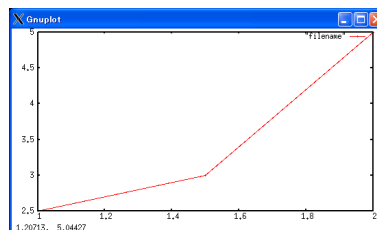
と打ち込むと上図のようなグラフが作成される。

「filename」というファイルに以下の数値がかかれていた場合

```
1.0      2.5
1.5      3.0
2.0      5.0
```

```
gnuplot > plot "filename" w lp
```

とすればファイルの数値が下図のようにプロットされる。「w lp」は点を線で結ぶオプションである。



| gnuplot の基本的なコマンド | 説明               |
|-------------------|------------------|
| plot              | 2次元プロット          |
| splot             | 3次元プロット          |
| set logscale x    | x 軸方向を対数プロット     |
| set logscale y    | y 軸方向を対数プロット     |
| set grid          | グリッドを表示          |
| set xlabel "ラベル名" | x 軸に名前を付ける       |
| set ylabel "ラベル名" | y 軸に名前を付ける       |
| set xrange[a : b] | x=a から x=b までを表示 |
| set yrange[a : b] | y=a から y=b までを表示 |