

FORTRANプログラムの相違点

山之上 卓*

1. はじめに

新システムの FORTRAN コンパイラ Sun FORTRAN の文法は、ANSI FORTRAN77 の拡張版であり、MVS や CMS で動作していたソースプログラムを書き換える必要は、一部のサブルーチンコール（数値計算ライブラリ、時間計測、グラフィックス等）等を除いて、ほとんどない。しかしながら、OS が異なるため、ソースプログラム名、コンパイルの方法、入出力ファイルの割当方法、オブジェクトプログラムの実行方法等を変更しなければならない。

2. ソースプログラム名

Sun FORTRAN では、ソースプログラムのファイル名の拡張子は、.f , .F または .for でなければならない。例えば MVS の

EXAMPLE.FORT

というファイル名は、

example.f

のように書き換えなければならない。

* 情報科学センター yamanoue@isct.kyutech.ac.jp

3. コンパイル

コンパイルは、コマンド

f77 [オプションの列] ソースファイル名, オブジェクトファイル名, ライブラリ名の列

で行う。

ソースファイル名には .f .F または .for の拡張子が付いていなければならない。

オブジェクトファイル名には .o の拡張子が付いていなければならない。

ディレクトリ /usr/lib に登録されているライブラリ lib____.a をリンクする場合には、ルートからのパス /usr/lib/lib____.a を指定する代わりに、-l____ を指定しても良い。

よく使用するオプションとして次のようなものがある。

-o 実行ファイル名

実行ファイル名を指示する。このオプションが省略された場合は、ファイル a.out に実行プログラムが出力される。

-c

コンパイルのみを行い、リンクしない。オブジェクトプログラムのファイル名には、.o の拡張子が自動的に付けられる。このオプションが省略された場合は、リンクまで行われる。

-O

最適化を行う。

例3. 1 ソースファイル ex1.f をコンパイルして a.out に実行ファイルを出力する場合。

f77 ex1.f

例3. 2 ソースファイル ex1.f をコンパイルして ex1 に実行ファイルを出力する場合.

```
f77 -o ex1 ex1.f
```

例3. 3 ソースファイル sub1.f をコンパイルしてオブジェクトファイル sub1.o を作成する場合

```
f77 -c sub1.f
```

例3. 4 ソースファイル sub2.f を最適化してコンパイルし、オブジェクトファイル sub2.o を作成する場合

```
f77 -O -c sub2.f
```

例3. 5 ソースファイル main.f をコンパイルし、オブジェクトファイル sub1.o と sub2.o をリンクし、ex2 に実行ファイルを出力する場合

```
f77 -o ex2 main.f sub1.o sub2.o
```

例3. 6 ソースファイル ex3.f を最適化してコンパイルし、IMSL の算術演算ライブラリ (mathライブラリ) をリンクし、ex3 に実行ファイルを出力する場合

```
f77 -O -o ex3 ex3.f -lmath
```

4. 入出力ファイルの割当と実行

READ 文や WRITE 文で使用される装置番号は、5 が標準入力ファイル stdin, 6 が標準出力ファイル stdout, 0 がエラー出力ファイル stderr に割り当てられる。

0, 5, 6 番以外の装置番号 n は、OPEN 文でファイルを割当ない限り、ファイル

fort.n

に割り当てられる。

例 4. 1

キーボード（標準入力ファイル）から2つの数を入力して、その和をディスプレイ（標準出力ファイル）に出力するプログラム ex1.f と、そのコンパイル及び実行例

プログラム ex1.f の内容

```
read(5,*) a,b
c=a+b
write(6,*) 'a+b=',c
stop
end
```

コンパイル及び実行

```
% f77 -o ex1 ex1.f
ex1.f:
MAIN:
% ex1
3.0, 4.0
a+b= 7.00000
%
```

例 4. 2

例 1 と同じプログラムで、ファイル input.data からデータを入力し、ファイル output.data へ実行結果を出力する場合.

```
% cat input.data    ... ファイル input.data の内容表示
   3.0, 4.0
```

```
% ex1 < input.data >output.data
   ... リダイレクションによって標準入出力をファイルに
   割り当てる
```

```
% cat output.data    ... ファイル output.data の内容表示
a+b=    7.00000
```

MVS のコマンドプロシジャや CMS の REXX に相当するものとして、UNIX にはシェルスクリプトがある。シェルスクリプトを記述することによって、ソースプログラムのコンパイル、ファイルの割当、実行をまとめて行うことができる。

例 4. 3

例 4. 1 で示したソースプログラムをコンパイルし、input.data からデータを入力させ、output.data に結果を出力させるシェルスクリプトおよびその実行例

シェルスクリプト ex1.sh の内容

```
#!/bin/sh
f77 -o ex1 ex1.f
ex1 <input.data >output.data
```

ex1.sh の実行

```
% chmod +x ex1.sh    ... ex1.sh を実行可能ファイルにする.
```

```
% cat input.data    ... ファイル input.data の内容表示
   3.0, 4.0
```

```
% ex1.sh             ... シェルスクリプトの実行.
ex1.f:
MAIN:
```

```
% cat output.data    ... ファイル output.data の内容表示
a+b=    7.00000
```

例 4. 4

0, 5, 6 番以外の装置番号を使用したプログラムと, そのコンパイル, ファイルの割り当ておよび実行を行うシェルスクリプト.

ソースプログラム ex2.f の内容

```
read(10,*) a,b
c=a+b
write(11,*) 'a+b=',c
stop
end
```

シェルスクリプト ex2.sh の内容

```
#!/bin/sh
f77 -o ex2 ex2.f
rm fort.10          ... 張られていたリンクをはずす
rm fort.11
ln -s input.data fort.10  ... リンクを張り直す
ln -s output.data fort.11
ex2
```

ex2.sh の実行

```
% chmod +x ex2.sh    ... ex2.sh を実行可能ファイルにする.

% cat input.data    ... ファイル input.data の内容表示
  3.0, 4.0

% ex2.sh            ... シェルスクリプトの実行.
ex2.f:
MAIN:

% cat output.data    ... ファイル output.data の内容表示
a+b=    7.00000
```

5. 簡易グラフィックス

UNIX は、標準で、Tektronix 4014 (以下 4 0 1 4) 等へグラフィックス出力を行うライブラリを持っている。また、Xウィンドウやパソコン用通信ソフト Kermit は、4 0 1 4 のエミュレータ機能を持っている。

ここでは、Xウィンドウや Kermit の 4 0 1 4 エミュレータ機能を利用する、FORTRAN の簡単なグラフィックスプログラムの書き方について説明する。

5. 1 plot サブルーチン

UNIX は以下のような、グラフィックス出力用サブルーチンを持っている。この一覧表は、「man 3 plot」を実行することによって表示される。

初期設定

openpl	: グラフィック出力用に画面を初期化する。
space(x1, y1, x2, y2)	: グラフィック領域の確保。

終了処理

erase	: グラフィックを消去する。
closepl	: グラフィック用の画面を閉じる。

描画ルーチン

point(x, y)	: (x, y) に点を描く。
line(x1, y1, x2, y2)	: (x1, y1) から (x2, y2) に直線を引く。
circle(x, y, r)	: (x, y) を中心に半径 r の円を描く。
arc(x, y, x1, y1, x2, y2)	: (x, y) を中心に (x1, y1) をから、 (x2, y2) まで、反時計方向に円弧をかく。
move(x, y)	: 始点をきめる。
cont(x, y)	: move(x, y) の始点から (x, y) まで直線を引く。
label(' string ')	: move(x, y) で指定された座標に文字列を描く。

注意: label 以外のサブルーチンの引数の型は、整数である。

プログラム例

```
*
* how2plot.f
*   a simple graphics example in FX/fortran
*   for Techtronics 4014 and 4014 emulators.
*
*   * in your kermit
*       just compile and execute.
*       [return] (return to shell)
*       in order to return vt102, you should do following sequence.
*       ^[c .... return to local kermit
*       Kermit-MS>set term vt102
*       Kermit-MS>c ... return to remote system
*       [return]
*
*   * in your X-window system, start one of following processes.
*       xterm -t or kterm -t
*
* How to Compile and execute
*   fortran -o how2plot -O how2plot.f -lfxg -l4014
*   how2plot
*
*   call openpl
*   call space(0,0,700,700)
*   move(x,y) ... move pen to (x,y)
*   call move(100,100)
*   cont(x,y) ... write line from preveious position to (x,y)
*   call cont(600,600)
*   line(x1,y1,x2,y2) ... write line (x1,y1)-(x2,y2)
*   call line(10,500,600,100)
*   arc(x-center,y-center,x-start,y-start,x-end,y-end) ... arc, c-clock
*   call arc(200,100,300,100,200,200)
*   circle(x,y,r) ... circle center:(x,y) radius:r
*   call circle(450,300,200)
*   point(x,y) ... point (x,y)
*   call point(100,400)
*   label('string') ... write string at last position
*   call move(300,100)
*   call label('plot example for 4014')
*   read(*,*)
*   call erase
*   call closepl
*   stop
*   end
```


5.2 コンパイル

コンパイルは

```
f77 [オプションの列] ソースプログラムの列 -lfxg -l4014 [他のライブラリの列]
```

で行う。ここで fxg は FORTRAN で plot サブルーチンを利用するための変換ライブラリであり、4014 は、テクトロニクス 4014 用の plot サブルーチン群である。

例 5.1 で示したプログラムのコンパイル例

```
% fortran -o how2plot how2plot.f -lfxg -l4014
```

5.3 実行

X ウィンドウで 4014 エミュレータを起動する場合は

```
% xterm -t      または      % kterm -t
```

を入力する。これで、4014 エミュレータのウィンドウが表示される。

4014 エミュレータのウィンドウ中で 5.2 で作成した実行ファイル名を入力すれば、グラフィックが表示される。5.2 の例で作成した実行ファイルの場合は、

```
% how2plot
```

を入力する。

プログラムを実行後、画面をクリアするには、

```
% clear
```

を入力する。

Kermit でグラフィックを表示する場合は、そのまま実行ファイル名を入力すれば良い。

自動的に4014のエミュレータとなり、グラフィックが表示される。Kermit で、画面をクリアし、端末モードを元に戻すには、

% gclr

を入力する。(コマンド gclr は、工学部物質工学科の長谷部教官が作成し、センターに提供していただいたものである。この場所を借りて御礼申しあげる。)

6. おわりに

FORTRAN プログラムの書き換え方ではなく、コンパイルと実行方法を中心に説明した。PHIGS などの高機能グラフィックスライブラリや IMSL ライブラリについては、また別の機会に説明を行う。

参考文献

- 1) Sun FORTRAN User's Guide
- 2) 溝口佳寛, 「Unixネットワーク入門」, インターフェース, No.151, pp.177-191, 1989-12.