

### 本日の授業のテーマ

今までの学習内容をまとめます。

- (1) 重要な UNIX コマンド
- (2) プログラムのコンパイルと実行
- (3) FORTRAN の基本命令と基本演算
- (4) プログラム例

本日の授業のゴールは、以下のとおり。

- UNIX の基本的なファイルの操作が理解できること。
- FORTRAN プログラムのコンパイルと実行方法が理解できること。
- プログラムの記述順序と実行順序が理解できる。
- FORTRAN の基本命令と基本演算が理解できること。
  - program 文
  - 変数宣言
    - 整数 INTEGER
    - 実数 REAL
  - キーボード入力 READ (5, \*)
  - ディスプレイ出力 WRITE (6, \*)
  - 基本演算
    - 代入文 =
    - 加算 +
    - 減算 -
    - 乗算 \*
    - 除算 /
    - べき乗 \*\*
  - STOP 文と END 文
- 注釈文が理解できること。
- 暗黙の型宣言が理解できること。

## 1 重要な UNIX コマンド

- UNIX のファイル構造は、ツリー(木)構造です。
  - カレントディレクトリーの 1 つ上のディレクトリーを親ディレクトリーと言います。親ディレクトリーへ移動するコマンドは、「`cd ..`」です。
  - カレントディレクトリーの直ぐ下のディレクトリーを子ディレクトリー、あるいはサブディレクトリーと言います。例えば、子ディレクトリー `hogehoge` に移動するコマンドは、「`cd hogehoge`」です。
- カレントディレクトリーにあるファイルやサブディレクトリーの名前を調べるコマンドは、「`ls`」です。
- サブディレクトリ `hogehoge` の追加と削除のコマンドは、以下の通り。
  - 追加は、「`mkdir hogehoge`」です。
  - 削除は、「`rmdir hogehoge`」です。
- ファイルを削除するコマンドは、「`rm hogehoge`」です。これで、`hogehoge` というファイルが削除されます。
- ファイルやディレクトリーをコピーするコマンドは、「`cp hogehoge hugahuga`」です。これで、`hogehoge` というファイルあるいはディレクトリーの `hugahuga` という名のコピーが作成されます。
- 以前使用したコマンドを呼び出す機能をヒストリー機能と言います。「↑」や「↓」で使用できます。同じような長いコマンドを何回も打ち込む手間が省けます。

## 2 コンパイルと実行

- FORTRAN のソースファイルの名前は、「`hogehoge.f`」のように拡張子「`.f`」が必要です。
- FORTRAN のソースファイル「`hogehoge.f`」をコンパイル(FORTRAN を機械語に翻訳)して、実行ファイル「`hugahuga`」を作成するコマンドは、次の通りです。

```
f77 -o hugahuga hogehoge.f
```
- ターミナルに「**実行ファイル名(例えば、`hugahuga`)**」を打ち込んで[Enter]キーを押せば、プログラムは実行されます。

## 3 FORTRAN の基本命令と基本演算

### 3.1 プログラムの記述順序と実行順序

- 制御文(今後学習する)が無い限り、プログラムは、上から下へ実行されます。
- そのプログラムを作成する場合の記述順序は、図 1 の通りです。実際の動作を記述する実行文に先立って、プログラム名(省略可能)と宣言文を書きます。

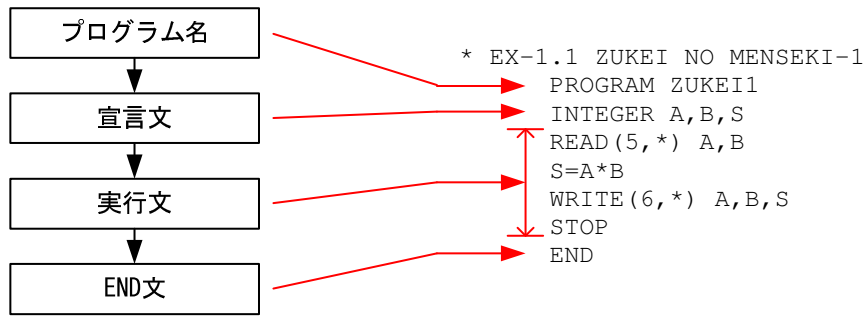


図1 左はプログラムの記述順序の流れ図、右はプログラムの例

### 3.2 PROGRAM 文

- 図1で示すように、「PROGRAM ZUKEI1」と書きます。
- 役割は、プログラムに名前を付けることです。図1では、「ZUKEI1」という名前になっています。これが無くても、プログラムのコンパイルと実行は可能です。

### 3.3 変数宣言

- 変数を使う場合、型名と変数名を書きます。実行文の前に宣言をする必要があります(図1)。したがって、一番最初を書くことになります。PROGRAM文がある場合は、その後に書きます。
- コンピューターの主な仕事は、データの処理です。データの中身は数字や文字です。このデータを処理する場合、その入れ物が必要です。その入れ物を変数と呼びます。
- 変数を使う場合、それに名前を付ける必要があります。入れ物の名前と考えてください。この名前をつける規則が決まっています(FORTRAN77)。
  - 変数名は、英文字で始まる6文字以内の英数字に限る。
  - 英文字は、A~Zのアルファベットです。
  - 英数字は、A~Zのアルファベットと0~9までの数字です。
- 秋田高専の計算機サーバーは機能が拡張されており、6文字以上の変数名も使えます。大文字と小文字の区別はありません。
- 数字と文字のデータでは、形が異なります。同じ数字でも実数と整数では、データの形が異なります。正確には、コンピューター内部での表現方法が異なります。データを処理する場合、それを入れる入れ物である変数が絶対に必要ですが、その入れ物も、データの形に合わせる必要があります。
- そのため、変数を使う場合、データの形を決める必要があります。これを型の宣言と言います。これまで、学習した型は、以下の通りです。

実数型	REAL
整数型	INTEGER

- 型の宣言文は、いくつあっても良いです。ただし、同じ変数名を2回宣言することは許されません。変数の型宣言の例は、以下の通りで、型名と変数名を書きます。

```
REAL XXY, ABC, T
INTEGER ZZ, AAA, KKK, I, J
```

### 3.4 キーボード入力

- キーボードから入力された数値を変数 A, B, C に格納する命令は、以下のように記述します。入力された順に、A, B, C に格納されます。

```
READ (5, *) A, B, C
```

- READ 文の括弧の最初の数値は、データの入力先を示します。5 は、FORTRAN 規格でキーボードと決められています。

### 3.5 ディスプレイ出力

- 変数 A, B, C の値をディスプレイ上書き出す命令は、以下のように記述します。WRITE 文で書かれた順に、A, B, C の値がディスプレイ上に 1 行で表示されます。

```
WRITE (6, *) A, B, C
```

- WRITE 文の括弧の最初の数値は、データの出力先を示します。6 は、FORTRAN 規格でディスプレイと決められています。

### 3.6 基本演算

- 等号(=)の記号を使いますが、その働きは、数学の等号とは異なります。FORTRAN での等号の働きは、右辺の算術式の値を左辺の変数に代入します。

FORTRAN 実行文	機能説明
A=B*C	B×C の演算結果を変数 A へ代入する
Z=3.1415	定数 3.1415 を変数 Z へ代入する

- 四則演算とべき乗の演算子は、以下の表(教科書 p.19 と同じ)の通りです。

演算子 (FORTRAN)	働き	例	数学的表現
+	加算	A+B	A+B
-	減算	A-B	A-B
*	乗算	A*B	A×B
/	除算	A/B	A/B 又は A÷B
**	べき乗	A**B	A <sup>B</sup>

### 3.7 STOP 文と END 文

- STOP 文によりプログラムの実行を止めます。
- END 文によって、プログラムの終わりを示します。

### 3.8 注釈行

- 注釈行(コメントラインとも言う)は、プログラムの内容をわかりやすくするために、書きます。第 1 桁目が、c または\*ならば、その行は注釈行です。

- END 文の前ならば、どこに入れても良いです。
- プログラムの実行には、全く影響を与えません。

### 3.9 暗黙の型宣言

- 変数は、型の宣言をしないうちでも使えます。その場合は、コンパイラが以下のように判断します。

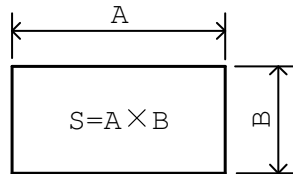
変数名の始まりが、I, J, K, L, M, N の場合	INTEGER
変数名の始まりが、それ以外の場合	REAL

## 4 プログラム例

教科書の p.4 と p.16 のプログラムの作成ができるようになってください。

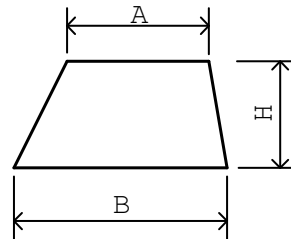
### 4.1 面積の計算 1 (教科書 p.4)

長方形の面積を求めるプログラムを書け。整数 A と B のデータをキーボードから読み込み、その面積 S をディスプレイに出力する。

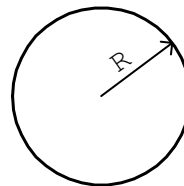


### 4.2 面積の計算 1 (教科書 p.16)

台形と円の面積を求めるプログラムを作れ。実数 A, B, H, R をキーボードから読み込み、面積 S1, S2 をディスプレイに出力する。



$$S1 = (A+B) \times H / 2$$



$$S2 = 3.1416 \times R^2$$

\* EX-1.1 ZUKEI NO MENSEKI-1

```
PROGRAM ZUKEI1
INTEGER A,B,S
READ(5,*) A,B
S=A*B
WRITE(6,*) A,B,S
STOP
END
```

\* EX-1.2 ZUKEI NO MENSEKI-2

```
PROGRAM ZUKEI2
REAL A,B,H,R,S1,S2
READ(5,*) A,B,H,R
S1=(A+B)*H/2
S2=3.1416*R**2
WRITE(6,*) A,B,H,R,S1,S2
STOP
END
```