

いまさら Fortran 入門

Introduction

- 惑星学では様々な局面で, プログラムを使った計算を行う.
 - データ解析, 理論計算, 数値実験 など.
- 今後履修する実習でも, データ解析や数値計算を行う(学ぶ)だろう. さらに, 卒業研究でも, 選択したテーマに依っては, プログラムを自分で作成し, それによるデータ解析や数値実験を多数実施することになるだろう.
- ここでは, 惑星学において良く使われるコンピュータ言語のひとつである Fortran を学ぶ.

Fortran その1

- Fortran (フォートラン) = Formula translation
- 科学技術計算で良く使われるコンピュータ高級言語のひとつ
 - 高級言語
 - (人間が)わかりやすい
 - ハードウェア(CPU やメモリ)の詳細を知らなくてよい
 - » 値を保持するためのメモリ上の場所(アドレス)などを考えなくてよい
 - Fortran 以外的高级言語
 - C, C++ など.
 - » なお, Unix は C 言語で書かれている.
 - 低級言語
 - 例えば, 機械語, アセンブラ
 - コンピュータがわかりやすい(～人間が分かりにくい)
 - ハードウェアの動作に近い命令ができる/命令をしなければならない

Fortran その2

- 科学技術計算に適している
 - Fortran は, 数式を数学の形式に近い形で表現できた初めてのプログラミング言語である.
- (比較的)簡単
 - Fortran は基本的には単純な言語であり, 修得が比較的容易である.
- 高速計算に優れるとされている
 - 言語が比較的簡単なため, プログラムを最適化(高速化)しやすい.
 - 近年では, C, C++ で書かれた大型計算のためのプログラムも増えている.
- 膨大な過去の資産がある
 - これまでに開発されて来た多数のプログラム・ライブラリがある.
 - それらの資産を利用することができる.

Fortran その3

- 現在でも様々に利用されている
 - 世界各国の数値天気予報
 - (日本の)気象庁, ヨーロッパ中期天気予報センター等
 - 世界各国の気候研究
 - 各国の気候研究センター, 大学等
 - N体計算
 - 宇宙空間プラズマ計算
- 注意
 - 業界によって良く使われるコンピュータ言語は異なるように見える. 上の例においても, Fortran 以外の言語が使われることはある.

Fortran の歴史

- 最初の Fortran コンパイラ開発 (1957 年)
- Fortran 66
 - 言語仕様が初めて標準化された。
- Fortran 77
 - 非常によく使われてきた。現在でも Fortran 77 で書かれたプログラムが使われている。
- Fortran 90
 - Fortran 77 に様々な改良を加えた後継言語
 - 自由形式, 再帰手続き, モジュール, ...
 - **本実習で主な対象とするバージョン**
- Fortran 95
- ...

Fortran プログラム実行への手順

- Fortran プログラムの実行には 3 ステップ
 - プログラム作成
 - エディタ(例えば, emacs)でプログラムを作成
 - コンパイル (compile) & リンク (link)
 - コンパイラ(例えば, gfortran)で, プログラミング言語で書かれたプログラム(ソースコード)を機械語(オブジェクトコード)に変換
 - コンパイラは, 必要に応じて, オブジェクトコードを必要なライブラリとリンクして実行ファイルを作成
 - 実行
 - 機械語で書かれたプログラム(実行ファイル)を実行する.
- Fortran はコンパイラ型言語
 - 上記のように, 機械語への変換と実行は分かれている.
 - 対義語はインタプリタ型言語
 - プログラムを機械語に変換しながら実行する
 - 変換しながら実行するため, 相対的に実行が遅い.
 - シェルスクリプト, ruby, perl, python, ...
 - ただし, コンパイラ型言語, インタプリタ言語の両方に対応できる言語も存在する.

Fortran プログラムの最低限決まり事

- ファイル名の末尾は “.f90” とする。
 - Fortran 77 で書かれたプログラム (正しくは, 「固定形式」で書かれたプログラム) のファイル名の末尾は “.f”
 - “.F90”, “.F” の末尾を持ったファイル名のプログラムもある. これらも Fortran のプログラム. 詳細は説明しない.
- プログラムの命令文は半角文字で書く。
 - Fortran では, コメント文と文字列の中身には全角文字を用いることができる. それ以外では全角文字を使ってはいけない.
- 大文字と小文字は区別されない。
 - 他の言語では, 大文字と小文字が区別されることが多い.
- 1 行は 132 文字まで。
 - より長い文を書く時には, 行の最後に “&” を書くことで, 次の行に継続される.
- ある行の “!” 以降はコメント文.
- プログラムで実行する命令は, 上から下に向かって順番に書く。
 - 逆に言えば, 命令文は上から下に向かって順に処理される.
 - 特別なことをしない限り, 逆に戻ることはない.

コンパイル (&リンク) 方法1

- エディタでプログラム(ここでは, hello.f90)を作ったら下の
ようにコンパイル (&リンク)する.

```
$ gfortran hello.f90
```

- 実習では, コンパイラに gfortran (GNU Fortran) を用いる.
 - 現在は有償・無償含めていくつかの Fortran コンパイラがある.
- 上記により, 実行ファイル a.out が作成される.
 - a.out は, Unix/Linux 上でのコンパイラの出力のデフォルトファイル名.
- 下のようにして実行.

```
$ ./a.out
```

コンパイル (&リンク) 方法2

- 下のようによりすることで, 実行ファイル名を指定することができる(ここでは hello).

```
$ gfortran -o hello hello.f90  
$ ./hello
```



ここで, 実行ファイル名 (hello) は, プログラム名 (hello.f90) とは完全に無関係に選ぶことができる.

プログラム作成時のアドバイス

- エラーメッセージを良く読む.
- 間違える(エラーが出る)ことに慣れる.
 - エラーが出るのは良くあることです.
 - 間違えたプログラムのエラーを一つずつなくしていくことが大切です.
 - いずれエラーのないプログラムになります.

実習

- 実習を通して, Fortran を使ってプログラミングを体験しましょう / プログラミングに慣れましょう.