© UNIVERSAL RESEARCH REPORTS | REFEREED | PEER REVIEWED
ISSN : 2348 - 5612 | Volume : 04 , Issue : 8 | October – December 2017

Study Of Program Structure And Different Data Types In Fortran

Sunita

Abstract : FORmula TRANslation. FORTRAN was developed in the 1950-s by a team of IBM programmers and released in 1957. It is a language special developed to let the computer execute number crunching operations. It was the first high level language, one in which the programmer did not have to directly deal with assembler or machine code (0's and 1's). It was the first language to be widely used and it is the oldest language still widely used today. It was designed to express scientific and mathematical formulas and it is still used in that area.

The general structure of a FORTRAN program contains :

PROGRAM name
Opening documentation
Variable declarations
Program statements and comments
END

- The PROGRAM statement (optional) gives a name to the program. The first character of the name must be a letter.
- The opening documentation tells the user what the program does, who wrote it, what utilities it uses (if any), dates written and revision history, etc.
- Variable declarations set up space in computer memory for the data types that will be used in the program.
- Program statements are the actual steps that solve the original problem. Comment lines are liberally included throughout to describe what the program is doing.
- The END statement terminates the program and returns control to the computer's operating system.

FORTRAN supports six different data types:

1. Integer
2. Real

3. Double precision
4. Complex
5. Character
6. Logical

The first four types are used for processing various types of NUMERIC data. Character type is used for processing STRINGS of characters. Logical type is for processing LOGICAL data; such data may have either .TRUE. or .FALSE. as its value.

The most often used data types in FORTRAN are INTEGER, REAL and CHARACTER we will focus on these in class. Each of these data types has two representations: they can be CONSTANTS or VARIABLES.

CONSTANTS

Constants are quantities whose values do not change through program execution. Do not confuse constants with variables that may have constant value.

INTEGER constants are a sequence of digits without commas or decimal points. A negative integer constant must be preceded by a minus sign, a plus sign is optional for positive integers.

VALID: 0 137 -2517 +3245

INVALID: 3,456 24.5 - - 5 7 -

REAL constants are numbers that have a decimal point without a comma. The rules for negative and positive signs also apply here.

VALID: 0. 12345.678 12.0 - .0001

INVALID: 0 1,234.99 22

REAL constants can also be represented in scientific notation where the exponent is raised to a power of ten and multiplied by the base. The following real number can be expressed in many ways:

Eg; 3.374 X 102 = 3.374E2

.3374E3

337.4E0

33740E-2

CONSTANT.

CHARACTER constants, or STRINGS, are sequences of characters or symbols enclosed within apostrophes (single quotes).

```
VALID:   'This is a STRING'  ' '   '&h"y^5)'
INVALID:  "  "OC3030"
```

VARIABLES

Variables are used in FORTRAN just as in any mathematical equation. FORTRAN defines certain rules for their use however.

Variable names must be between 1 and 32 letters or numbers in length with the first character being a letter.

```
VALID:   A   AB   Z1   MASS   LENGTH  ABC123  NOTTOOLONG
INVALID: 1VAR   R2-D2   A*AB4   6FEET
```

Variables must be one of the six types discussed earlier. The type of the variable determines what kind of data can be assigned to it.

Variables then, must be declared at the top of the program before any executable statements.

```
REAL  ALPHA,  BETA,  KOUNT
INTEGER  Z, A1
CHARACTER*8  HGT, STRING*13, L9*2
```

NOTE: For CHARACTER variables, the LENGTH of the string is defined.

If a variable is not explicitly declared, it will be assigned a type based on the default naming convention. This convention states that the variable is:

```
INTEGER   if the first letter is  I,J,K,L,M,N  (I-N)
REAL      any other letter
```

References :

1. https://www.tutorialspoint.com/fortran/fortran_constants.htm
2. http://www.mathcs.emory.edu/~cheung/Courses/561/Syllabus/5-Fortran/variables.html
3. http://www.infis.univ.trieste.it/fortran/constant.html

4. http://www.oc.nps.edu/~bird/oc3030_online/fortran/basics/basics.html