Programación en FORTRAN

Nivel Básico - Sesión 2

Martin Josemaría Vuelta Rojas

29 de diciembre de 2017

SoftButterfly

Contenido

Variables y tipos de datos

Variables y tipos de datos

- Un lenguaje de programación permite identificar los datos que se manipulan y almacenan en grandes cantidades en un ordenador.

Variables y tipos de datos

- Un lenguaje de programación permite identificar los datos que se manipulan y almacenan en grandes cantidades en un ordenador.

Variables

Variables y tipos de datos

- Un lenguaje de programación permite identificar los datos que se manipulan y almacenan en grandes cantidades en un ordenador.

Variables

- Una variable es un objeto que representa un tipo de dato, suceptible de modificarse, nombrado por cadenas de caracteres.

Tipos de datos

Variables y tipos de datos

Tipos de datos

1. **character:** cadena de uno o varios caracteres.

Variables y tipos de datos

Tipos de datos

1. **character:** cadena de uno o varios caracteres.
2. **integer:** números enteros, positivos o negativos.

Variables y tipos de datos

Tipos de datos

1. **character:** cadena de uno o varios caracteres.
2. **integer:** números enteros, positivos o negativos.
3. **logical:** valores lógicos o booleanos, es decir, toman uno de los dos valores, .true. (verdadero) o .false. (falso).

Variables y tipos de datos

Tipos de datos

1. **character:** cadena de uno o varios caracteres.
2. **integer:** números enteros, positivos o negativos.
3. **logical:** valores lógicos o booleanos, es decir, toman uno de los dos valores, .true. (verdadero) o .false. (falso).
4. **real:** números reales, positivos o negativos.

Variables y tipos de datos

Tipos de datos

1. **character:** cadena de uno o varios caracteres.
2. **integer:** números enteros, positivos o negativos.
3. **logical:** valores lógicos o booleanos, es decir, toman uno de los dos valores, .true. (verdadero) o .false. (falso).
4. **real:** números reales, positivos o negativos.
5. **complex:** números complejos, compuestos de una parte real y otra imaginaria, ambas de tipo real.

Variables y constantes

Declaración de variables

- La declaración de una o más variables del mismo tipo está dada por la sintaxis

  <tipo> ,[<atributo(s)>] [::] <variable(s)> [=<valor>]

Variables y constantes

Declaración de variables

- La declaración de una o más variables del mismo tipo está dada por la sintaxis

  <tipo> ,[<atributo(s)>] [::] <variable(s)> [=<valor>]

- Algunos atributos son:
  parameter, save, pointer, target, allocatable, dimension, public, private, external, intrinsic, optional.

Variables y constantes

Declaración de variables

- La declaración de una o más variables del mismo tipo está dada por la sintaxis

  <tipo> ,[<atributo(s)>] [::] <variable(s)> [=<valor>]

- Algunos atributos son:
  parameter, save, pointer, target, allocatable, dimension, public, private, external, intrinsic, optional.

```
1   CHARACTER(len= 4), PARAMETER :: prompt = ">>> "
2   CHARACTER(len= *), PARAMETER :: message = "Ingresa tu primer nombre [máx 20 car]:"
```

Variables y constantes

Declaración de variables

- La declaración de una o más variables del mismo tipo está dada por la sintaxis

  <tipo> ,[<atributo(s)>] [::] <variable(s)> [=<valor>]

- Algunos atributos son:
  parameter, save, pointer, target, allocatable, dimension, public, private, external, intrinsic, optional.

```
1  CHARACTER(len= 4), PARAMETER :: prompt = ">>> "
2  CHARACTER(len= *), PARAMETER :: message = "Ingresa tu primer nombre [máx 20 car]:"
```

Véase strings.f95

Declaración de constantes

- Si se requiere que una variable que tome un valor definido no suceptible de cambio, se utiliza el atributo parameter.

```
1    CHARACTER, PARAMETER :: NewLine = CHAR(10)
```

Variables y constantes

Declaración de constantes

- Si se requiere que una variable que tome un valor definido no suceptible de cambio, se utiliza el atributo parameter.

```
1   CHARACTER, PARAMETER :: NewLine = CHAR(10)
```

Véase strings.f95

Declaración de constantes

- Si se requiere que una variable que tome un valor definido no suceptible de cambio, se utiliza el atributo parameter.

```
1   CHARACTER, PARAMETER :: NewLine = CHAR(10)
```

Véase strings.f95

- Las variables pueden ser definidas en función de constantes mediante el atributo parameter.

Cadenas de caracteres

Declaración de cadenas de caracteres

- La declaración de una variable de tipo character está dada por la sintaxis
  character[(len=<longitud>)],[<atributo(s)>][::]<variable(s)>[=<valor>]

Cadenas de caracteres

Declaración de cadenas de caracteres

- La declaración de una variable de tipo character está dada por la sintaxis
  character[(len=<longitud>)],[<atributo(s)>][::]<variable(s)>[=<valor>]

```
1   CHARACTER(kind=ascii, len=26) :: Alphabet
2   CHARACTER(kind= ucs4, len=30) :: HelloWorld
```

Cadenas de caracteres

Declaración de cadenas de caracteres

- La declaración de una variable de tipo character está dada por la sintaxis
  character[(len=<longitud>)],[<atributo(s)>][::]<variable(s)>[=<valor>]

```
1   CHARACTER(kind=ascii, len=26) :: Alphabet
2   CHARACTER(kind= ucs4, len=30) :: HelloWorld
```

Véase kind_character.f95

Valores lógicos

asd

Enteros

Tipos de enteros

-

Reales

Tipos de reales

Reales

Tipos de reales

-

Reales

Tipos de reales

-
- La sintaxis para el tipo real es

real(kind=<np>)

Reales

Tipos de reales

-
- La sintaxis para el tipo real es

real(kind=<np>)

```
1   REAL(kind=p04) :: X04
2   REAL(kind=p08) :: X08
3   REAL(kind=p16) :: X16
4   REAL(kind=p32) :: X32
```

Reales

Tipos de reales

-
- La sintaxis para el tipo real es

real(kind=<np>)

```
1   REAL(kind=p04) :: X04
2   REAL(kind=p08) :: X08
3   REAL(kind=p16) :: X16
4   REAL(kind=p32) :: X32
```

Véase kind_real.f95