

Ejercicios dinámica de Fluidos

Johan S. Méndez

2018/09/6

1. Tipos de formato en salida de datos, Fortran 90

Cuando presentamos los resultados numéricos es posible ser mucho más preciso a la hora de dar la información a través del programa. Para hacer esto se usan las declaraciones **FORMAT**. A través del uso del formato se puede hacer lo siguiente

- Especificar cuántas columnas presentará el programa
- Especificar dónde estará el punto decimal
- Especificar dónde habrá un punto en blanco
- Títulos

La instrucción **FORMAT** es la que nos permite manipular la forma de los datos de salida del programa

1.1. Enteros i

Este formato es de los más simples que podemos encontrar y de los más comunes, indicando solamente que el dato de salida será entero. Por ejemplo **i3**.^{es} un entero que toma tres columnas. Consideremos el siguiente ejemplo

```
program prueba
integer :: T
  print *, "Tabla de 12 espacios"
  print *, " "
  do t = 1, 12
    print 100, t, t*12
    100 format(" ", i3, "*12=", i3)
  end do
end program prueba
```

Prestemos atención a la instrucción **FORMAT**

```
100 format(" ", i3, "*12=", i3)
```

Dentro de los paréntesis tenemos

- Imprima lo que está dentro de las comillas, en este caso es un espacio en blanco
- la coma separa los items en la instrucción
- **i3** Imprime la primera variable justificando a la derecha
- separador de item
- **"*12="** Imprime lo que se encuentra dentro de las comillas
- separador de item

- **i3** Imprime la primera variable justificando a la derecha

1.2. Reales f

El formato **f** puede entenderse como una extensión del formato **i**, La diferencia es que el formato **f** especifica en donde se va a colocar el decimal y cuantas cifras significativas queremos poner. Entonces, **f7.4** significa

- Hay una cantidad total de siete
- Este es el punto decimal
- Cuatro dígitos después del punto decimal

El formato **f7.4** es usado para números en el rango

$$-9,9999 < x < 99,9999$$

Cuando el programa intente escribir fuera de este rango presentará siete estrellas, aludiendo a un error en la definición en el formato de salida de datos. Miremos un ejemplo que tiene ambos tipos de formato

```
program prueba
implicit none
integer :: fluid
real :: litres
real :: pints
do fluid = 1,10
    litres = fluid/1.75
    pints = fluid*1.75
    print 100, pints, fluid, litres
    100 format(" ", f7.3, " ", i3, " ", f7.3)
end do
end program prueba
```

1.3. Reales e

La notación científica o también llamada exponencial es de ayuda para algunos casos prácticos en la elaboración de programas. Si necesitamos los datos en un rango grande usaremos este formato. Por ejemplo se utiliza **e10.4**, el número 10 da los lugares de las cifras significativas, el número ubicado después del decimal da las posiciones que escribirá después del punto. Tomemos un ejemplo concreto, 1000 puede ser escrito como 10^{*3} o como $0.1 \cdot 10^{*4}$. Esto consta de dos partes: 0.1 da las cifras significativas, mientras que 10^{*4} da el exponente. En Fortran podemos escribirlo como $,1E+04$, donde $E+04$ representa 10^{*4}

En general debemos tener en cuenta los siguientes aspectos

- El punto decimal
- El signo del número entero
- El signo del exponente
- La magnitud del exponente
- E

A continuación un ejemplo del uso de este formato

```

program prueba
implicit none
integer :: i
real :: small = 1.0
real :: big = 1.0
do i = 1,50
    print 100,i,small,big
    100 format(" ", i3," ",e10.4," ", e10.4)
    small = small/10.
    big = big*10.0
end do
end program prueba

```

1.4. Espacios x

Tenemos en fortran dos manera de generar espacios a la hora de la salida de datos, la primera de ellas es poner espacios dentro de las comillas, como lo mencionaba anteriormente, la segunda forma es usa la instrucción **x**, consideremos el soiguiente fragmento de código

```

print 100, alpha, beta
100 format(1x,f10.4,10x,f10.3)

```

El 10x indica 10 espacios en blanco

Para finalizar se muestra un pequeño programa que mezcla todas las salidas de datos, de esta manera se pueda entender de mejor manera la utilización de cada uno de estos formatos

```

program prueba
implicit none
character (len = 15) :: FirstName
integer :: age
real :: weight
character :: weight
character (len = 1) :: sex
print *, "Escriba su primer nombre"
read *, FirstName
print *, " Escriba su edad en años"
read *, age
print *, "Escriba su peso en kg"
read *, weight
print *, "escriba su sexo (F/M)"
read *, sex
print *, "Tus datos personales son : "
print *
print 100
100 format(4x,"Primer Nombre",4x,"edad",1x, "Peso",2x,"sex")
print 200, FirstName, age, weight,sex
200 format(1x,a,2x,i3,2x,f5.2,2x,a)
end program prue

```