



Física Computacional

Escuela de Física

M.R.Fulla¹

¹Escuela de Física, Universidad Nacional de Colombia Sede
Medellín

marlonfulla@yahoo.com- Oficina:21-408

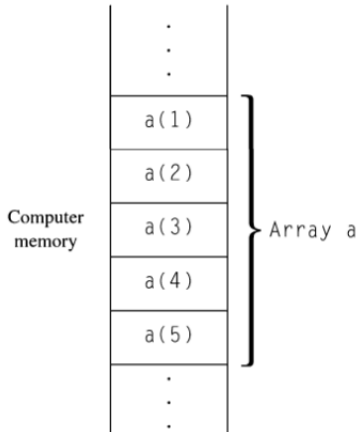
<https://sites.google.com/view/fiscomunalmed/>

August 31, 2023

Arreglos

Asignación

TYPE, DIMENSION(int-val)::var



Definiciones

Rango : número de subíndices
Grado : número de elementos
Forma : rango y grado
Tamaño : número total de elementos

```
9      PROGRAM arreglos
10     REAL, DIMENSION(10)::temp
11     REAL, DIMENSION(10,10)::temp_vs_tiempo
12     END PROGRAM arreglos
```

Arreglos con valores particulares

```
9      PROGRAM arreglos
10     REAL, DIMENSION(3)::constantes=(/3.141592,2.718281,0.70716/)
11     WRITE(*,*) 2*constantes(1)*constantes(3)
12     END PROGRAM arreglos
```

Inicialización de Arreglos

1. En la declaración (/val1,val2,.../)
2. Por asignación

```
9      PROGRAM arreglos
10     REAL, DIMENSION(10)::tiempo
11     INTEGER::i
12     DO i=1,10,1
13         tiempo(i)=i
14     END DO
15     END PROGRAM arreglos
```

3. Con un DO-IMPLÍCITO (arg1,arg2,index=istart,iend,inc)

```
9      PROGRAM arreglos
10     REAL, DIMENSION(10)::tiempo
11     INTEGER::i
12     tiempo=/(i,i=1,10,1)/
13     END PROGRAM arreglos
```

4. Con READ

```
9      PROGRAM arreglos
10     REAL, DIMENSION(10)::tiempo
11     INTEGER::i
12     DO i=1,10,1
13     WRITE(*,*) "Escribe la entrada: ",i
14     READ(*,*) tiempo(i)
15     WRITE(*,*) tiempo(i) !ECHO
16     ENDDO
17     END PROGRAM arreglos
```

5. Inicialización con Operación Conjunta

```
9      PROGRAM arreglos
10     REAL, DIMENSION(10)::tiempo
11     tiempo=0.0
12     END PROGRAM arreglos
```

Cambio del Rango de Subíndices

REAL,DIMENSION(-100:100)::Temperatura

Fuera del límite

```
9      PROGRAM arreglos
10     REAL, DIMENSION(10)::tiempo
11     tiempo=0.0
12     WRITE(*,*) tiempo(11)
13     END PROGRAM arreglos
```

100 % No issues found

Error List

Entire Solution 2 Errors 0 Warnings 0 Messages Build + IntelliSense

Code	Description
	Compilation Aborted (code 1)
	error #5560: Subscript #1 of the array TIEMPO has value 11 which is greater than the upper bound of 10

Nota: Declarar el tamaño de un array con un parámetro es altamente recomendable:

INTEGER, PARAMETER::tamano=10

REAL, DIMENSION(tamano)::tiempo

Arreglos

Operaciones Conjuntas

Dos arreglos con la misma forma → operaciones aritméticas ordinarias

a(1)	1	b(1)	5.	d(1)	6.
a(2)	2	b(2)	6.	d(2)	8.
a(3)	3	b(3)	7.	d(3)	10.
a(4)	4	b(4)	8.	d(4)	12.
a		+	b	=	d

```
9      PROGRAM arreglos
10     IMPLICIT NONE
11     REAL, DIMENSION(4)::a,b,c
12     INTEGER::i
13     a=(/1.,2.,3.,4./)
14     b=(/5.,6.,7.,8./)
15     !REALICEMOS LA OPERACIÓN c=a-2*b
16     !Método 1, sumando por componentes
17     DO i=1,4
18       c(i)=a(i)-2*b(i)
19     ENDDO
20     WRITE(*,*)(c(i),i=1,4,1) !IMPRIMIMOS CON DO IMPLÍCITO
21     !Método 2, sumando por componentes
22     c=0.0 !Reinicializamos a 0
23     c=a-2*b
24     WRITE(*,*)(c(i),i=1,4,1) !IMPRIMIMOS CON DO IMPLÍCITO
25     END PROGRAM arreglos
```

```
"C:\Users\cjdea\source\repos" X + v
-9.000000    -10.00000    -11.00000    -12.00000
-9.000000    -10.00000    -11.00000    -12.00000
Press any key to continue . . .
```

$\text{abs}(a-2*b) \rightarrow$ función elemental intrínseca

Subconjunto de un Arreglo

Sección del arreglo que en la forma más general, se puede construir através de una tripleta de subíndices:

subíndice1: subíndice2: paso

```
9      PROGRAM arreglos
10     IMPLICIT NONE
11     REAL, DIMENSION(10)::a
12     REAL, DIMENSION(4)::b
13     INTEGER::k
14     a=(/1.,2.,3.,4.,5.,6.,7.,8.,9.,10./)
15     b=a(2:8:2)
16     WRITE(*,*)(b(k),k=1,4) !IMPRIMIMOS CON DO IMPLÍCITO
17     b=a(:8:2)
18     WRITE(*,*)(b(k),k=1,4) !IMPRIMIMOS CON DO IMPLÍCITO
19     END PROGRAM arreglos
```

2.000000	4.000000	6.000000	8.000000
1.000000	3.000000	5.000000	7.000000

Press any key to continue . . .

otras tripletas posibles:

subíndice1 : subíndice2

subíndice1 :

subíndice1 : : paso

: subíndice2

: subíndice2 : paso

: : paso

:

- ▶ Escribir un programa que defina en el encabezado, el grado \mathbb{R}^n através de un número (n) de dos vectores y reciba por consola sus componentes. El programa debe retornar su producto punto.
- ▶ Escribir un programa que reciba por consola las componentes de dos vectores en \mathbb{R}^3 y retorne su producto cruz.

Nota: tenga en mente todas las buenas prácticas de programación.