

Programación básica

FORTTRAN. ARREGLOS
MULTIDIMENSIONALES

¿Dónde estamos en el ciclo de vida del software?



Sets de datos (multidimensionales)

Hemos utilizado un tipo de “recipiente” más sofisticados para la contener información durante la vida de un programa. Ahora extenderemos ese concepto a tener varias dimensiones en estos “recipientes”



Arreglos multidimensionales

Los arreglos pueden ser unidimensionales (como se vió en el tema anterior) o multidimensionales. Nos da una capacidad de organización y acceso a un set de datos con un solo identificador

Vegetales

Marduk

Recipiente 1,1

Recipiente 2,1

Recipiente 3,1

Recipiente 4,2



Recipiente 1,2

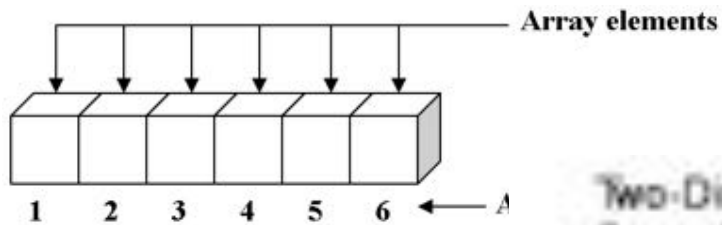
Recipiente 2,3

Recipiente 3,4

Recipiente 4,5

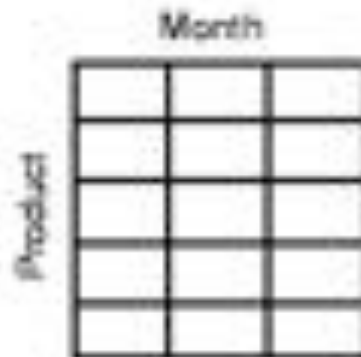


Arreglos multidimensionales



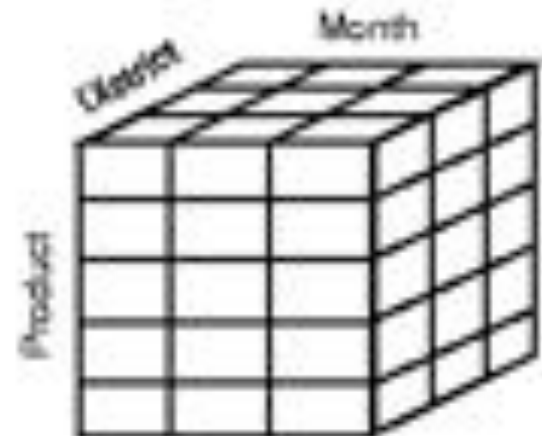
One-dimensional array with six

Two-Dimensional Spreadsheet



Sales data for each district is in a separate spreadsheet

Multidimensional Array



Sales data for all districts is in a single array

Declaración en Fortrán y uso

```
program arregloBidimensional
```

```
c Este programa genera un arreglo bidimensional de  
c 3 renglones y 3 columnas y accede a cada elemento  
c del arreglo a través de un ciclo do
```

```
integer i,j, matriz(3,3)
```

```
matriz(1,1) = 1
```

```
matriz(1,2) = 2
```

```
matriz(1,3) = 3
```

```
matriz(2,1) = 4
```

```
matriz(2,2) = 5
```

```
matriz(2,3) = 6
```

```
matriz(3,1) = 7
```

```
matriz(3,2) = 8
```

```
matriz(3,3) = 9
```

```
write (*,*) 'Imprimir matriz'
```

```
do i = 1, 3, 1
```

```
do j = 1, 3, 1
```

```
write (*,*) matriz(i,j)
```

```
enddo
```

```
enddo
```

```
stop
```

```
end
```

j (columnas)

(1,1) 1	(1,2) 2	(1,3) 3
(2,1) 4	(2,2) 5	(2,3) 6
(3,1) 7	(3,2) 8	(3,3) 9

i (renglones)

Equivalencia entre dimensiones

j (columnas)

i (renglones)

(1,1) 1	(1,2) 2	(1,3) 3
(2,1) 4	(2,2) 5	(2,3) 6
(3,1) 7	(3,2) 8	(3,3) 9

Multidimensional

(1) 1	(2) 2	(3) 3	(4) 4	(5) 5	(6) 6	(7) 7	(8) 8	(9) 9
-----------------	-----------------	-----------------	-----------------	-----------------	-----------------	-----------------	-----------------	-----------------

Unidimensional

Manejo de los índices

Si no lo especificamos los índices (dimensión) de un arreglo se manejaran automáticamente del 1 al valor que le demos en la dimensión. Estos índices se pueden modificar.

`integer matriz(3,3)`

i (renglones)	j (columnas)		
	(1,1) 1	(1,2) 2	(1,3) 3
	(2,1) 4	(2,2) 5	(2,3) 6
	(3,1) 7	(3,2) 8	(3,3) 9

`Integer, dimension(-3:-1 ,3) :: matriz`

i (renglones)	j (columnas)		
	(-3,1) 1	(-3,2) 2	(-3,3) 3
	(-2,1) 4	(-2,2) 5	(-2,3) 6
	(-1,1) 7	(-1,2) 8	(-1,3) 9

Operaciones suma o resta matrices

$$A = \begin{pmatrix} 2 & 0 & 1 \\ 3 & 0 & 0 \\ 5 & 1 & 1 \end{pmatrix} \quad B = \begin{pmatrix} 1 & 0 & 1 \\ 1 & 2 & 1 \\ 1 & 1 & 0 \end{pmatrix}$$

$$A + B = \begin{pmatrix} 2+1 & 0+0 & 1+1 \\ 3+1 & 0+2 & 0+1 \\ 5+1 & 1+1 & 1+0 \end{pmatrix} = \begin{pmatrix} 3 & 0 & 2 \\ 4 & 2 & 1 \\ 6 & 2 & 1 \end{pmatrix}$$

```
program restaDeMatrices
```

```
c Este programa genera las matrices (A y B) y después  
c las resta generando una tercera matriz C
```

```
integer i,j,cont  
integer, dimension (-4:-1,3) :: matrizA, matrizB  
integer matrizC(-4:-1,3)
```

```
c se genera la matriz A
```

```
cont = 1  
do i = -4, -1, 1  
  do j = 1, 3, 1  
    matrizA(i,j) = cont  
    cont = cont + 1  
  enddo  
  write (*,*) ''  
enddo
```

```
c se genera la matriz B
```

```
do i = -4, -1, 1  
  do j = 1, 3, 1  
    cont = cont - 1  
    matrizB(i,j) = cont  
  enddo  
  write (*,*) ''  
enddo
```

```
c se imprimen las matrices generadas
```

```
write (*,*) 'Matriz A'                      Matriz B'  
do i = -4, -1, 1  
  do j = 1, 3, 1  
    write(*,*) matrizA(i,j), ' ', matrizB(i,j)  
  enddo  
  write (*,*) ''  
enddo
```

```
c se genera e imprime la matriz C
```

```
write (*,*) 'Matriz C = Matriz A - Matriz B'  
do i = -4, -1, 1  
  do j = 1, 3, 1  
    matriz(i,j) = matrizA(i,j) - matrizB(i,j)  
    write (*,*) matrizC(i,j)  
  enddo  
  write (*,*) ''  
enddo
```

```
stop  
end
```