



# *Arreglos*

---



# Arreglo

---

## *Concepto*

*Una variable arreglo es una estructura de datos y representa un conjunto homogéneo de datos identificados por un único nombre y almacenados en posiciones contiguas de memoria.*

Cada componente del arreglo puede ser accedida de manera directa, sin necesidad de recorrer los componentes anteriores.

Un arreglo ocupa una cantidad fija de memoria, determinada por la cantidad y tipo de sus componentes. Por lo que se la considera una estructura de datos estática.



# Arreglo

---

## *Declaración de un arreglo*

Para declarar una variable de tipo arreglo, se indica el nombre de la variable, precedido del tipo de datos de sus componentes y seguido del tamaño del arreglo encerrado entre corchetes.

Ejemplo:

```
real venta[30]
```

También puede declararse, definiendo previamente el tamaño del arreglo a través de una constante:

```
Constante N=30  
real venta [N]
```

En forma general, la declaración de un arreglo lineal o unidimensional:

**<tipo> <identificador-arreglo> [<tamaño>]**



# Operaciones con Arreglos

---

- Carga: Colocar información en el arreglo.
- Recorrido: Una vez cargado, se recorre el arreglo para mostrar o procesar su información.
- Copia: pasar el contenido de un arreglo a otro.
- Búsqueda: Buscar un elemento en el arreglo.
- Ordenamiento: clasificar los elementos del arreglos de acuerdo a un criterio.



# Carga de Arreglos

---

Una forma de ingresar los datos al arreglo de N componentes es por medio de la lectura desde teclado de los N valores, de la siguiente manera:

Para i Desde 0 Hasta N-1

Leer venta[i]

FinPara

La carga de los elementos de un arreglo también puede realizarse por medio de asignaciones:

Para i Desde 0 Hasta N-1

venta[ i ] = 0

FinPara



# Carga de Arreglos

---

También pueden cargarse las componentes de un arreglo de manera aleatoria, ya sea por lectura o asignación de un valor.

Por ejemplo:

entero pos

real valor

Escribir "ingrese la posición de la componente y el valor"

Leer pos

Leer valor

venta [pos -1]=valor //las posiciones empiezan en 1, el índice de las componentes en 0



# Recorrido de un Arreglo

---

Si se desea escribir alguna componente del arreglo se debe colocar el identificador de la estructura, seguido del su índice, encerrado entre corchetes.

Para i Desde 0 Hasta N -1

Escribir `venta[ i ]`

FinPara

# Seguimiento con Arreglos

Algoritmo **Ventas**

constante N=10

**void carga ( real v[N])**

Comienzo

entero i

Para i Desde 0 Hasta N-1

    Escribir " Ingresa Ventas ", i+1

    Leer v[i]

FinPara

retorna()

Fin

**void muestra (real Ven[N]), real prom)**

comienzo

entero i

    Escribir " Ventas realizadas:"

    Para i Desde N-1 Hasta 0

        Entonces "Importe de la venta " i+1, " es" Ven[i]

    FinPara

retorna ()

Fin

**Comienzo /\*---Algoritmo principal---\*/**

real venta[N]

carga (venta)

muestra (venta)

**Fin**

| Subprograma carga |   |        |
|-------------------|---|--------|
| v                 | i | Salida |
|                   |   |        |
|                   |   |        |
|                   |   |        |
|                   |   |        |
|                   |   |        |

| Subprograma muestra |   |        |
|---------------------|---|--------|
| Ven                 | i | Salida |
|                     |   |        |
|                     |   |        |
|                     |   |        |
|                     |   |        |
|                     |   |        |

| Ventas |  |  |  | Salida |
|--------|--|--|--|--------|
|        |  |  |  |        |
| 100h   |  |  |  |        |





# Carga Aleatoria

---

## **Algoritmo Importe\_Aleatorio**

void **carga\_aleat** ( real imp[36])

Comienzo

real importe

entero nc, i

Para i Desde 0 Hasta 49

    Escribir "Ingrese número de cajero e importe recaudado"

    Leer nc, importe

    imp [**nc - 1**] = importe

FinPara

retorna

Fin

\*\*\* Algoritmo Principal \*\*

Comienzo

real impor[50] ,

**carga\_aleat(impor)**

    .....

    .....

Fin



# Operaciones básicas con Arreglos

| Inicio                           | Proceso  | Fin  |
|----------------------------------|--|--|
| CONTADOR<br>Se crea el arreglo   | En un subprograma , dentro de una iteración se cuenta:<br>$C[\text{indice}] = C[\text{indice}] + 1$                | En un subprograma se usa el arreglo de contadores. |
| ACUMULADOR<br>Se crea el arreglo | En un subprograma , dentro de una iteración se acumula:<br>$C[\text{indice}] = C[\text{indice}] + \text{Variable}$ | En un subprograma se usa el arreglo                |



# Operaciones básicas con Arreglos

| Inicio                  | Proceso   | Fin  |
|-------------------------|---|--|
| MAXIMO<br><br>Max=0     | Si Max < Arre[indice]<br>Entonces<br>Max = Arre[indice]// actualiza<br>[si se guardan valores<br>relacionados se usan<br>variables auxiliares]<br>finsi | El valor mayor encontrado es <b>único</b> , pero puede repetirse o no.<br>En caso de repetirse se hace un subprograma Compara, donde se recorre el arreglo y<br>Si Max == Arre[indice]<br>Entonces<br>Se hace un proceso o se escribe un mensaje.<br>finsi |
| MINIMO<br><br>Min=99999 | Si Min > Arre[indice]<br>Entonces<br>Min = Arre[indice]// actualiza<br>[si se guardan valores<br>relacionados se usan<br>variables auxiliares]          | El valor menor encontrado es <b>único</b> , pero puede repetirse o no.<br>En caso de repetirse se hace un subprograma Compara, donde se recorre el arreglo y<br>Si Min == Arre[indice]<br>Entonces<br>Se hace un proceso o se escribe un mensaje.<br>finsi |



# Operaciones básicas con Arreglos

| Inicio                         | Proceso   | Fin  |
|--------------------------------|---|--|
| BANDERA<br><br>bandera = falso | Se identifica un evento,<br>analizando los datos del arreglo<br>en la condición.<br>Si CONDICION<br>entonces bandera = verdadero<br>sino indice = indice + 1 [se<br>sigue recorriendo el<br>arreglo]<br><br>finsi | Se evalúa la bandera. Si sucede<br>el evento devuelve verdadero,<br>sino habrá recorrido todo el<br>arreglo y devuelve falso.<br><br>Si bandera == verdadero<br>entonces // EL EVENTO SUCEDIÓ<br>sino // EL EVENTO NO SUCEDIÓ<br>finsi |