



## Ciclo 1

# Semana 6

## Estructuras de Datos Lineales

---

### Lectura 1 – Operaciones con Arreglos Unidimensionales



## | Operaciones con Arreglos Unidimensionales

Las operaciones que se pueden realizar con estos arreglos son las siguientes:

- **Asignación:** a cada elemento se le puede asignar un valor, según el tipo del arreglo, por ejemplo si el arreglo NUMEROS es de tipo entero, se podría asignar al elemento en la posición cero el valor 5 de la siguiente forma: **NUMEROS[0] = 5**. El valor del índice dado entre **[ ]**, puede ser un número entero positivo o una expresión aritmética que genere un número entero positivo.
- **Recorrido:** un arreglo se puede recorrer para introducir (escribir) datos en él o para obtener su contenido (leer). Estas operaciones se realizan utilizando estructuras repetitivas, con una variable de control entera que actúe como índice del vector, iniciando en cero y terminando en **N-1**, si N es el tamaño del vector.
- **Lectura/escritura:** para leer valores desde el teclado y asignarlos al vector se debe definir un ciclo que recorra uno a uno los elementos del vector y le asigne el valor leído desde el teclado, por ejemplo:

*n = tamaño del arreglo.*

*X: arreglo [n] entero*

*i = 0*

*Mientras i < n hacer*

*Leer (valor)*

*X[i] = valor*

*i = i + 1*

*Fin-mientras*



## Semana 6

## Estructuras de datos lineales

Para mostrar los valores almacenados en un arreglo, también se debe definir un ciclo que recorra uno a uno los elementos del vector y muestre en pantalla el valor de cada uno de ellos. Por ejemplo:

$n$  = tamaño del arreglo.

$X$ : arreglo  $[n]$  entero

$i = 0$

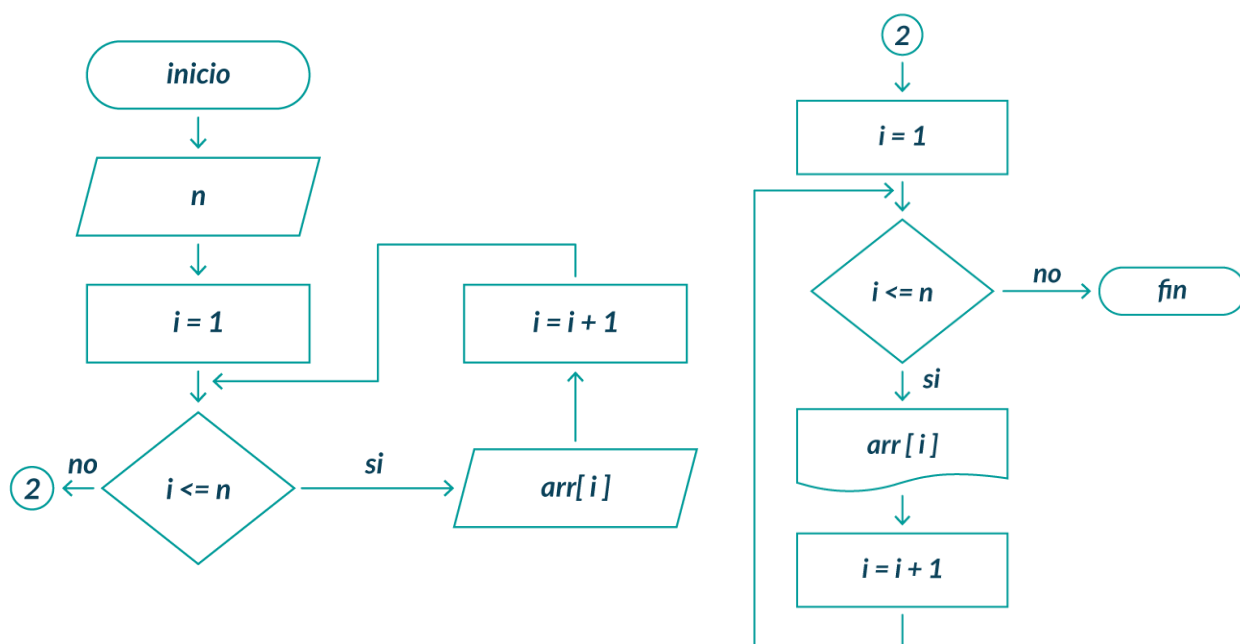
Mientras  $i < n$  hacer

Mostrar( $X[i]$ )

$i = i + 1$

Fin-mientras

En la siguiente gráfica se muestra el diagrama de flujo de un algoritmo que primero lee y asigna uno a uno los valores a un arreglo (**arr**) y luego muestra uno a uno los elementos del arreglo (**arr**), en este ejemplo (tomado de slidshare) **el arreglo (arr)** tiene sus posiciones desde 1 hasta  $n$ .





## Semana 6

## Estructuras de datos lineales

- **Ordenamiento:** hay varios algoritmos para realizar el ordenamiento de un vector, se debe recorrer todo el arreglo y hacer comparaciones con los elementos de tal forma que el arreglo quede ordenado. Ver las siguientes animaciones:
  - **Menú con los ordenamientos**  
[http://lwh.free.fr/pages/algo/tri/tri\\_es.html](http://lwh.free.fr/pages/algo/tri/tri_es.html)
  - **Ordenamiento por burbuja**  
[http://lwh.free.fr/pages/algo/tri/tri\\_bulle\\_es.html](http://lwh.free.fr/pages/algo/tri/tri_bulle_es.html)
  - **Ordenamiento por inserción**  
[http://lwh.free.fr/pages/algo/tri/tri\\_insertion\\_es.html](http://lwh.free.fr/pages/algo/tri/tri_insertion_es.html)
  - **Ordenamiento por selección**  
[http://lwh.free.fr/pages/algo/tri/tri\\_selection\\_es.html](http://lwh.free.fr/pages/algo/tri/tri_selection_es.html)
  - **Ordenamiento shell**  
[http://lwh.free.fr/pages/algo/tri/tri\\_shell\\_es.html](http://lwh.free.fr/pages/algo/tri/tri_shell_es.html)
  - **Quick-sort**  
[http://lwh.free.fr/pages/algo/tri/tri\\_rapide\\_es.html](http://lwh.free.fr/pages/algo/tri/tri_rapide_es.html)
- **Búsqueda:** Se recorre el vector posición por posición, comparando cada elemento del arreglo con el elemento buscado, iniciando en cero e incrementando en uno la posición, hasta que el vector se termina o se encuentra el elemento buscado. Para ampliar más el tema se puede ver:

[http://artemisa.unicauca.edu.co/~nediaz/EDDI/cap02.htm#ancla2\\_2](http://artemisa.unicauca.edu.co/~nediaz/EDDI/cap02.htm#ancla2_2).