APOYA

Introducción a la Programación

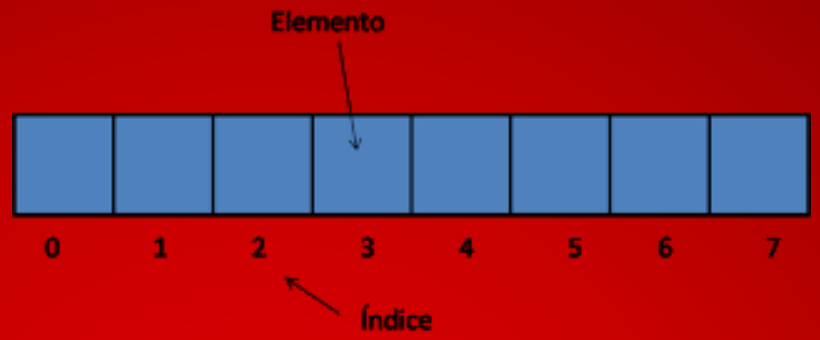
Comisión “B”

Año 2020

Arreglos: Vectores

Profesor: Ing. Gabriel Guismin

ARREGLOS



- **Un arreglo es un conjunto finito, ordenado y homogéneo de elementos.**

- ***Ordenado*** significa que se puede identificar al primer elemento, al segundo, al tercero y así sucesivamente.

- ***Finito***, significa que la cantidad de esos elementos será conocida y fija.

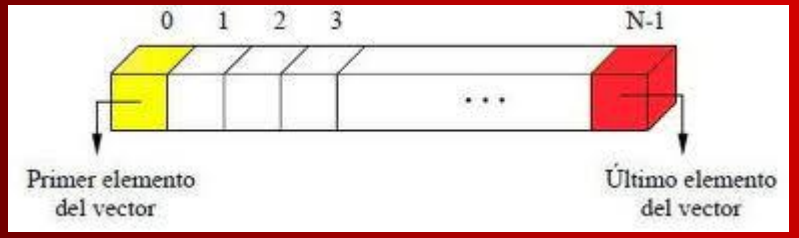
- ***Homogéneo***, indica que todos los elementos son del mismo tipo de datos.

ARREGLOS

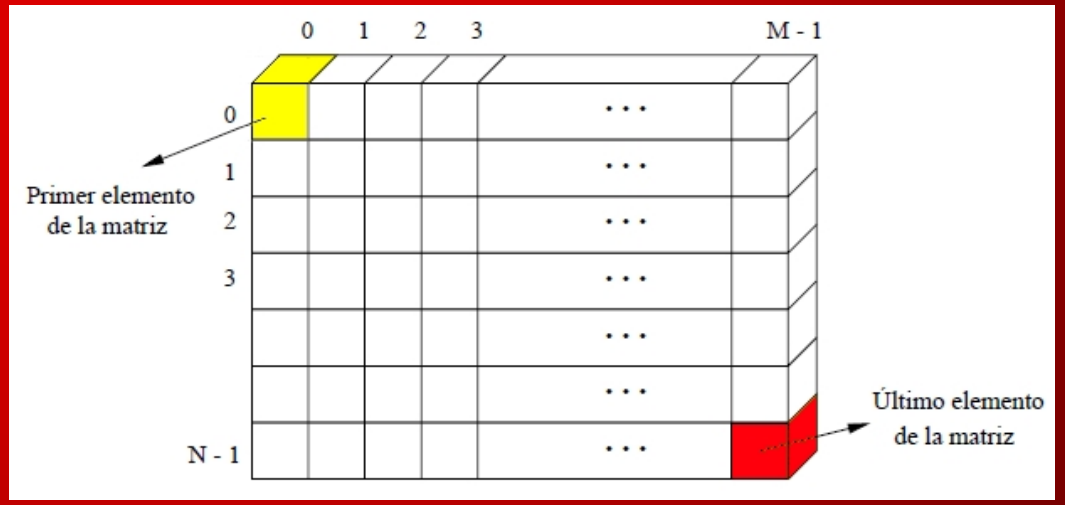
- Existen dos tipos de arreglos muy utilizados:

los unidimensionales (llamados **vectores**) y;

los bidimensionales (llamados **matrices**)



Declaración de un arreglo unidimensional - VECTOR



Declaración de un arreglo bidimensional - MATRIZ

Declaración de un Arreglo

```
<tipo de datos> nombre[dimensiones]
```

Declaración de un arreglo

- Por ejemplo, sea un arreglo de una sola dimensión (vector) llamado **datos** con 20 (veinte) elementos, todos del tipo entero:

```
entero datos[20]
```

Declaración de un vector con 20 elementos

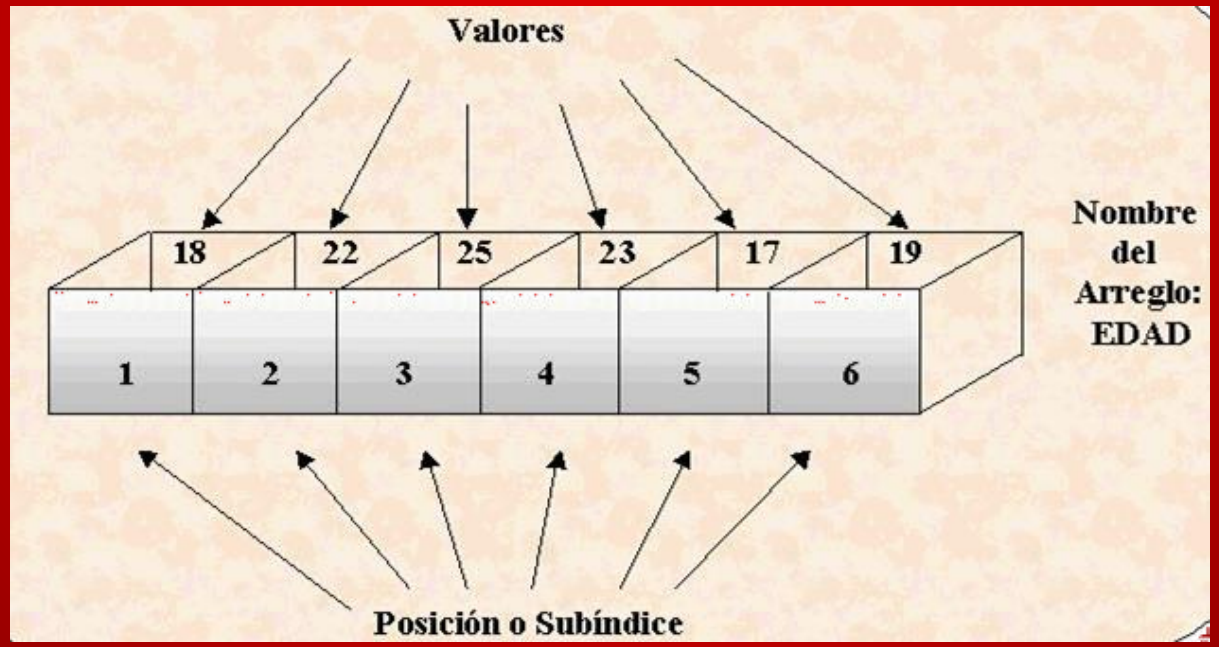
```
Dimension vector[5]
```

Declaración de un vector en PSeInt

Ing. Gabriel Guismin

Vectores

Un vector es un tipo de arreglo cuya particularidad es que posee una sola dimensión.
Podría decirse que el vector es el tipo de arreglos más simple.
Un vector es un arreglo lineal de datos con un solo índice; es decir, de una sola dimensión.



Vectores: Asignación

Si se declara un **vector[10] ,** donde se almacenarán datos enteros.

Es válido:

vector[1] = 5
vector[5] = 33
vector[9] = 33

NO es válido:
vector[11] = 7

El índice hace referencia a una ***celda inexistente*** (el vector fue declarado con diez celdas, no existe la celda de índice once). Se dice que esto es un error por **desbordamiento** y produce la interrupción instantánea en la ejecución del algoritmo.

Vectores: Lectura y Escritura

La **lectura** de datos en el vector y la **escritura** de los datos almacenados se pueden realizar directamente indicando las celdas involucradas

```
Leer vector[1]
Leer vector[2]

Escribir vector[1]
Escribir vector[2]
```

Ejemplo de Lectura y Escritura en posiciones indicadas de un vector

Ing. Gabriel Guismin

Vectores: Acceso secuencial (recorrido)

Recorrer una estructura significa acceder a todos los elementos que la conforman en forma secuencial (o a una parte de ella). Esto se realiza utilizando una estructura de control repetitiva.

```
para i ←1 hasta 10
     escribir (datos[i])
finpara
```

Ejemplo recorrido de un vector con la estructura PARA

```
i ←1
mientras(i < 11)
     escribir (datos[i])
     i ←i + 1
finmientras
```

Ejemplo recorrido de un vector con la estructura MIENTRAS

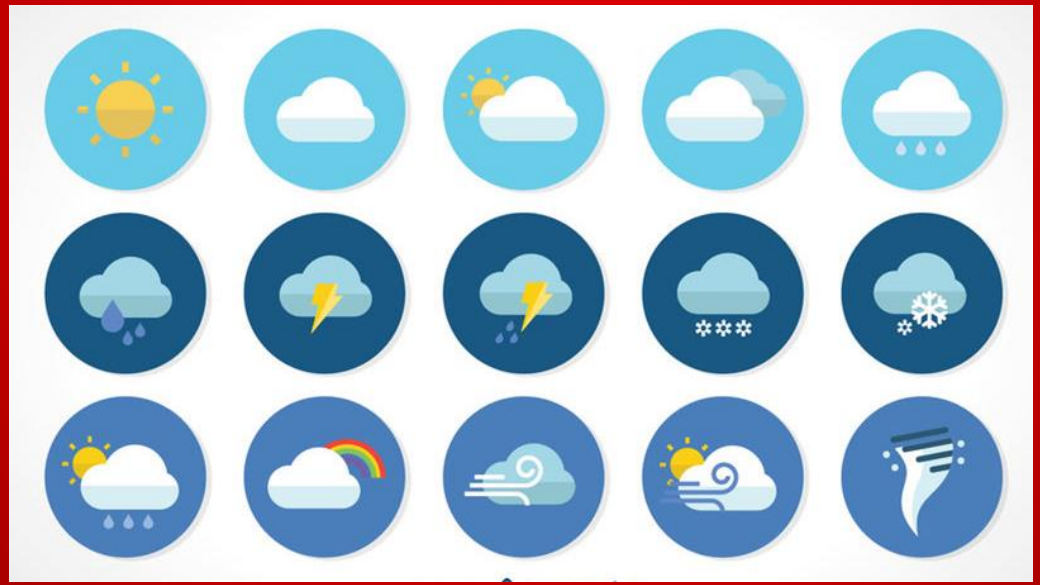
```
i ←1
repetir
     escribir (datos[i])
     i ←i + 1
hasta que (i = 11)
```

Ejemplo recorrido de un vector con la estructura REPETIR

Ing. Gabriel Guismin

Vectores: Ejemplo Práctico

Problema: Se necesita un algoritmo para almacenar las temperaturas máximas de los 31 días del mes de Enero pasado e informar el día más caluroso. Las temperaturas serán leídas.



Ing. Gabriel Guismin

Para conocer: en JAVA

```
tipo_dato nombre_array[];
nombre_array = new tipo_dato[tamanio];
```

Declaración e inicialización de un arreglo en Java

```
char arrayCaracteres[];
arrayCaracteres = new char[10];
```

Ejemplo: Declaración e inicialización de un arreglo en Java

```
char array[];
array = new char[10];

for (int i=0;i<array.length;i++)
  System.out.println(array[i]);
```

Ejemplo de recorrido de un arreglo en Java

Ing. Gabriel Guismin