



USMP

FACULTAD DE
INGENIERÍA Y ARQUITECTURA

Semestre Académico 2022-I



ASIGNATURA

Algoritmo y Estructura de Datos I

Docente: Ing. Juan Puerta Arce

**Escuela Profesional de
Ingeniería de Computación y Sistemas**

Semana 4

- Arreglos: Definición y clasificación.
- Arreglos unidimensionales: Declaración y creación.
- Formas de almacenamiento de datos.
- Visualización del contenido de un Vector.



USMP

FACULTAD DE
INGENIERÍA Y ARQUITECTURA

Arreglos



Introducción

Hasta ahora, sólo habíamos almacenados datos en variable de tipo int, boolean, double, char o String / string.

Sin embargo, **el uso de variables no es suficiente para resolver problemas mas complejos.**

Justificación

Situación: si se necesitase registrar los nombres y apellidos de 40 trabajadores de una empresa necesitaríamos declarar 40 variables.

```
String nomape1;  
String nomape2;  
String nomape3;  
...  
String nomape40;
```

Si la empresa tuviera N cantidad de empleados se requerirían N cantidad de variables.



Solución

La **solución** para casos parecidos a la situación anterior es usar **ARREGLOS**.

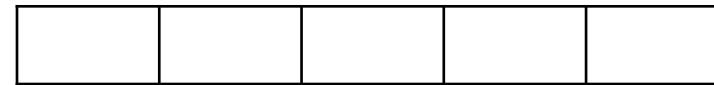
Definición

- Un arreglo es una colección de valores que tienen el mismo tipo de dato. A estos valores se les denomina elementos.
- Los arreglos son estructuras de datos que almacenan datos del mismo tipo. Los arreglos son entidades estáticas en cuanto a que su tamaño no cambia una vez que han sido creadas.
- Un arreglo es un conjunto de porciones de memoria RAM contiguas, del mismo tipo de datos y con el mismo nombre, y estas porciones de memoria se diferencian por un índice. El arreglo es una entidad estática en cuanto a que su tamaño no cambia una vez que han sido creadas.

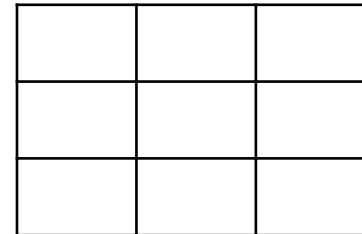
Clasificación

- Según el número de dimensiones que tenga, se clasifican en:

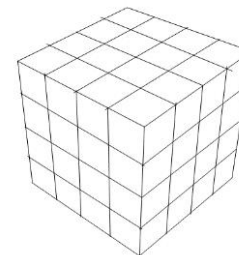
- Unidimensionales (Vectores)



- Bidimensionales (Matrices)



- Multidimensionales





USMP

FACULTAD DE
INGENIERÍA Y ARQUITECTURA

Vectores

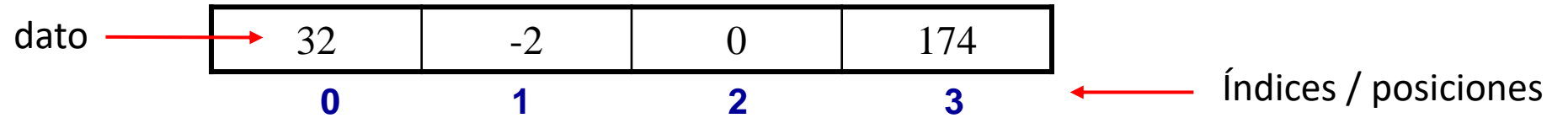
Características

- Los elementos (datos) de un Vector se almacenan en posiciones sucesivas de memoria y todos del mismo tipo de dato.
- A estos elementos se acceden a través de un número entero positivo llamado índice, el cual representa la posición de un elemento dentro del Vector.
- El tamaño del vector esta determinado por la cantidad de elementos (datos) que se pueden almacenar.
- El primer elemento (dato) se ubica en la posición 0.
- El último elemento (dato) se ubica en la posición *TamañoDelVector* - 1

Ejemplos de vectores

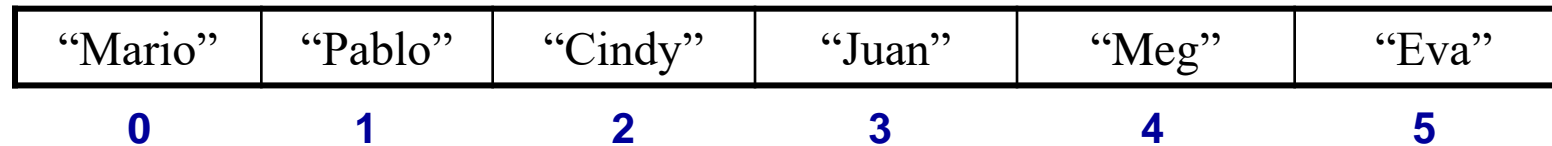
numeros[]

int



tamaño = 4

nombres[]
String / string



tamaño = 6

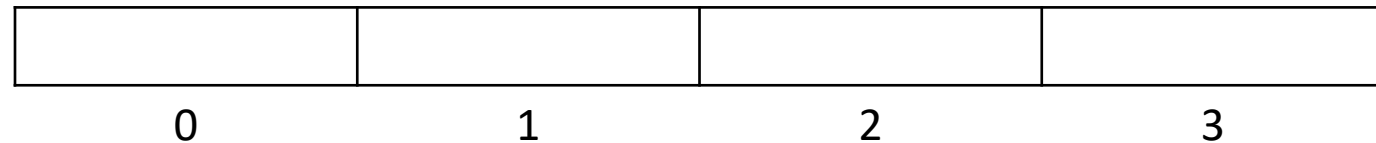
Creación de vectores

Sintaxis:

```
tipo_de_dato[ ] nombreDelVector;  
nombreDelVector = new tipo_de_dato[tamaño];
```

Ejemplo:

```
int[ ] notas;  
notas = new int[4];
```



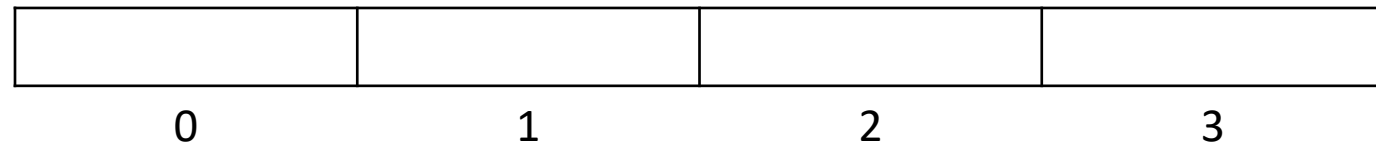
Creación de vectores

Sintaxis:

```
tipo_de_dato[ ] nombreDelVector = new tipo_de_dato[tamaño];
```

Ejemplo:

```
int[] notas = new int[4];
```



Almacenar un dato en un vector

Sintaxis:

NombreDelVector[posición] = valor;

Ejemplo:

notas[0] = 14;

notas[1] = 5;

14	5		
0	1	2	3

Acceder a datos almacenados en un vector.

Sintaxis:

NombreDelVector[*posición*]

Ejemplo:

notas[0]

notas[1]

14	5		
0	1	2	3



Operaciones con vectores

- Recorrido
- Búsqueda
- Modificación
- Ordenamiento
- Inserción
- Eliminación



USMP

FACULTAD DE
INGENIERÍA Y ARQUITECTURA

Recorrido de un Vector

Recorrido

Consiste en “**apuntar**” a una **posición o índice** del Vector y de manera secuencial avanzar a las siguientes.

vEjemplo[0] <----- se “apunta” a la posición 0 del vector

vEjemplo[1] <----- se “apunta” a la posición 1 del vector

vEjemplo[2] <----- se “apunta” a la posición 2 del vector

vEjemplo[3] <----- se “apunta” a la posición 3 del vector

...

vEjemplo[N] <----- se “apunta” a la posición N del vector

vEjemplo



0

1

2

...

...

N

TamañoDelVector - 1

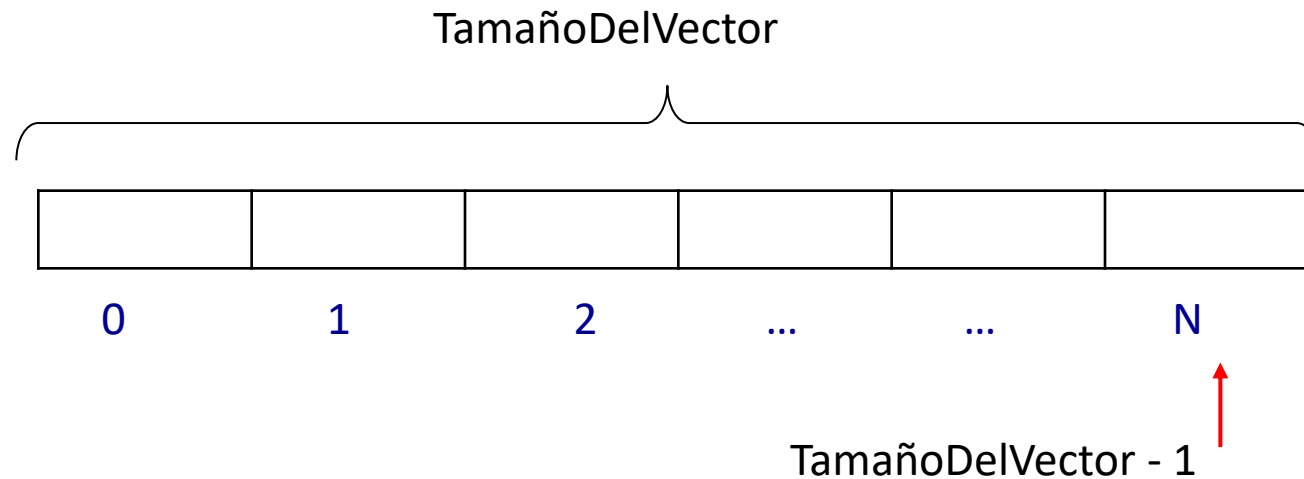


Recorrido con la sentencia for

Consiste en “**apuntar**” a una **posición o índice** del Vector y de manera secuencial avanzar a las siguientes usando la instrucción **for**.

```
for (int i = 0 ; i < TamañoDelVector; i++) {  
    ...  
    ... nombre_del_vector[i] ...  
    ...  
}
```

La variable **i** hace que el valor de la posición sea dinámico





Registro de datos total y parcial

Registro parcial de datos

- Es cuando los datos son ingresados en diferentes momentos, y no necesariamente el vector queda lleno.
- Se deben usar sentencias distintas al FOR.

Registro total de datos

- Es cuando los datos son ingresados de inicio a fin en un "mismo momento".
- Por lo general se usa la sentencia for.

Origen de los valores que se pueden almacenar en un vector.

- Existe al menos 3 formas en las cuales se puede almacenar valores en un Vector:
 - Al momento de escribir el programa.
 - Valores generados en forma aleatoria o calculada.
 - Valores ingresados durante la ejecución del programa.



GRACIAS.



USMP

FACULTAD DE
INGENIERÍA Y ARQUITECTURA

Escuela Profesional de
Ingeniería de Computación y Sistemas

Acreditada por:

