

Implementación

Elementos de C#

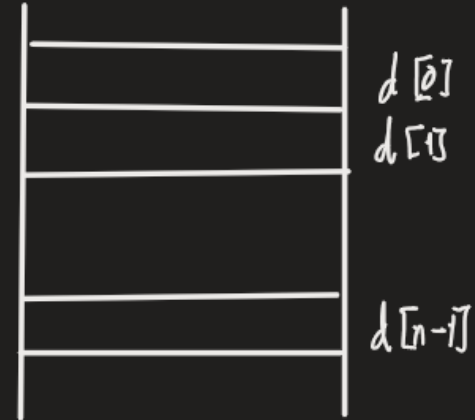
Arreglos

Sea un conjunto de datos

$$d = \{d_1, d_2, \dots, d_n\}$$

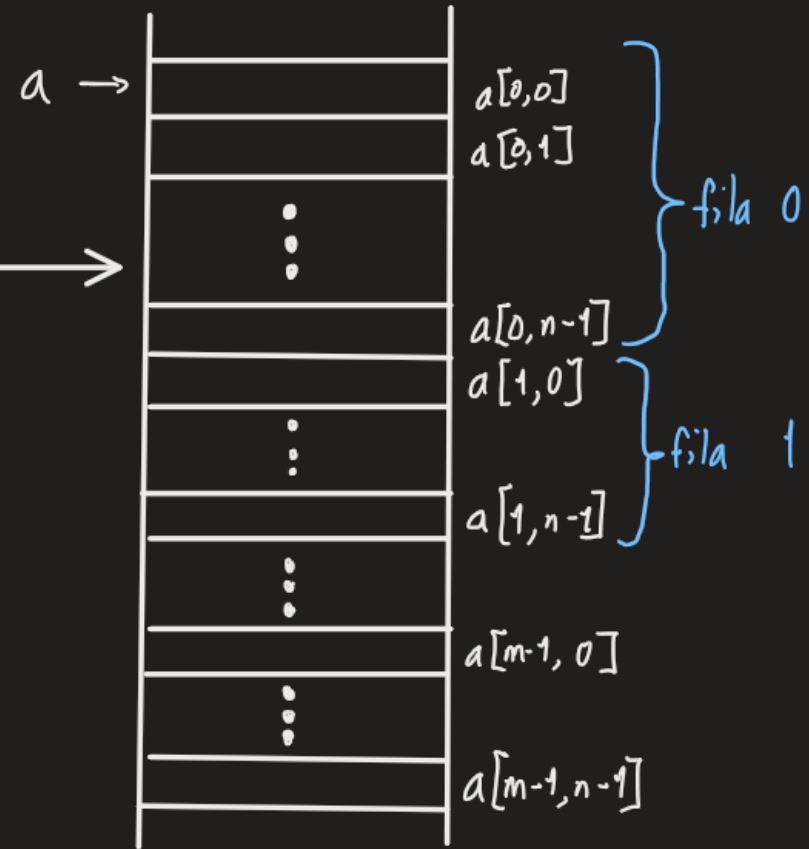


$$d = \{d[0] \dots d[n-1]\}$$



$$a = \begin{bmatrix} a_{11}, a_{12}, \dots \\ a_{21}, a_{22}, \dots \\ \vdots \\ a_{n1}, a_{n2}, \dots \end{bmatrix}$$

$$\begin{bmatrix} a_{00}, a_{01}, \dots, a_{0n-1} \\ a_{10}, a_{11}, \dots, a_{1n-1} \\ \vdots \\ a_{m-1,0}, a_{m-1,1}, \dots, a_{m-1,n-1} \end{bmatrix}$$



- 1.- Un arreglo siempre es adyacente en mas elementos
- 2.- Todos los elementos son de un solo tipo
- 3.- El nombre del arreglo es la referencia al inicio del arreglo
- 4.- En C# los arreglos son dinamicos

Definición de un arreglo

tipo [] d; // 1er orden

tipo [,] a; // 2 dimensiones

tipo [,,] b; // 3 dimensiones

tipo [][] c; // arreglo irregular

Definición de un arreglo

tipo $[T] d$; // 1 dim

tipo $[,] a$; // 2 dim

tipo $[,,,] b$; // ≥ 3 dim

tipo $[,,,] c$; // arreglo irregular

Instancia de arreglos (operador memoria)

$d = \text{new tipo}[n]$; $n, m \in \mathbb{N}_c$

$a = \text{new tipo}[m, n]$;

$b = \text{new tipo}[m, n, k]$;

$c = \text{new } \{ \text{new tipo} \}_{n \times k}$
 $\quad \quad \quad \text{new tipo} \}_{m \times k}$
 $\quad \quad \quad \{ \}$

$\begin{bmatrix} 1 & 2 & 3 \\ 1 & 2 & 3 & 4 \\ 1 & 2 & 2 & 4 \end{bmatrix}$

Para manejar los arreglos,
se utiliza una iteración

$\forall i = 0, n-1$
 $d[i]$

$\forall i = 0, m-1$
 $\forall j = 0, n-1$
 $a[i, j]$

\vdots
 0

Ejemplos

```
int[] x = new [] { 1, 2, -2, 4 };
```

{ 1, 2, -2, 4 }

```
double[] y = new [] { 1.1, -2.5, 4.3 };
```

{ 1.1, -2.5, 4.3 }

```
double[,] a = new [2, 3] {  
    { -1.1, 2.1, 3.1 },  
    { 2.1, 1.1, 3.1 },  
};
```

$\begin{bmatrix} -1.1 & 2.1 & 3.1 \\ 2.1 & 1.1 & 3.1 \end{bmatrix}$

```
int[][] b = new [] {  
    new [] { 1, 2, 3 },  
    new [] { 1 },  
    new [] { 1, 2, 3, 4 },  
};
```

$\begin{bmatrix} 1 & 2 & 3 \\ 1 \\ 1 & 2 & 3 & 4 \end{bmatrix}$

PRÁCTICA

Terminando poniendo a la salida los valores de los arreglos utilizando Console.WriteLine

```
→ for (int i = 0; i < n; i++)  
    {  
        Console.WriteLine(x[i]);  
    }
```

Ejemplos

```
int[] x = new [] { 1, 2, -2, 4 };
```

```
{ 1, 3, -2, 4 }  
{ 1.1, -2.5, 4.3 }
```

```
double[] y = new [] { 1.1, -2.5, 4.3 };
```

```
double[,] a = new [,] {  
    { -1.1, 2.1, 3.1 },  
    { 2.1, 1.1, 3.1 },  
};
```

```
[ -1.1  2.1  3.1 ]  
[  2.1  1.1  3.1 ]
```

```
int[][] b = new [] {  
    new [] { 1, 2, 3 },  
    new [] { 1 },  
    new [] { 1, 2, 3, 4 },  
};
```

```
[ 1  2  3 ]  
[ 1 ]  
[ 1  2  3  4 ]
```

PRÁCTICA

Terminar poniendo a la salida los valores de los arreglos utilizando `Console.WriteLine`

```
→ for (int i = 0; i < n; i++)  
    {  
        Console.WriteLine(x[i]);  
    }
```