





# ARREGLOS UNIDIMENSIONALES

**SEMANA 12**

# AGENDA



III	<b>Arreglos</b> Al finalizar la unidad, el estudiante implementa algoritmos para resolver problemas utilizando métodos de caso, formularios y arreglos unidimensionales y bidimensionales en el lenguaje C#; demostrando lógica y optimización de código en sus programas, y aplicando un sistema de control de versiones para gestionar su desarrollo y facilitar la colaboración.	11	Arreglos unidimensionales: Operaciones: Registrar, mostrar, buscar, modificar (D. FLUJO o pseudocódigo y código)
		12	Arreglos unidimensionales Operaciones: Insertar, eliminar, ordenar (burbuja) (D. FLUJO o pseudocódigo y código)
		13	Taller de desarrollo de casos <b>Evaluación T3</b>
		14	Cadenas de Caracteres. Funciones de manipulación de cadenas de caracteres Arreglo de cadenas. Arreglos paralelos. Ordenamientos. (D. FLUJO o pseudocódigo y código).
		15	Arreglos bidimensionales (Matrices) Operaciones con matrices (D. FLUJO o pseudocódigo y código)
		16	<b>EVALUACIÓN FINAL</b>



Al término de la clase, el estudiante elabora  
códigos en C# usando arreglos  
unidimensionales mostrando una lógica  
coherente al implementar su algoritmo

# ARREGLOS DE UNA DIMENSIÓN



Un array (arreglo) es una estructura de datos que nos permite almacenar un conjunto de datos del mismo tipo. El tamaño de los arrays se declara al inicio y no puede cambiar.

Se declara así:

**TipoDato** variable[ **tamaño** ];

```
1 using System;
2 using System.Collections.Generic;
3 using System.Linq;
4 using System.Text;
5 using System.Threading.Tasks;
6
7 namespace AppClase11 {
8     class Program {
9         static void Main(string[] args) {
10             string[] nombres = new string[4];
11
12             nombres[0] = "Carlos";
13             nombres[1] = "Julian";
14             nombres[2] = "Cristian";
15             nombres[3] = "Miguel";
16
17             //nombres[4] = "Miguel"; //error porque el índice máximo es 3
18
19             Console.WriteLine($"Zero: {nombres[2]}");
20         }
21     }
22 }
```

Posición  
al leer o  
escribir

Cantidad de  
cajones al  
declarar



# ARREGLOS DE UNA DIMENSIÓN



Programa un arreglo con 5 elementos solicitados al usuario y muestre su suma:

```
1  using System;
2  using System.Collections.Generic;
3  using System.Linq;
4  using System.Text;
5  using System.Threading.Tasks;
6
7  namespace AppClase11 {
8      class Program {
9          static void Main(string[] args) {
10              int[] num = new int[5];
11              int sum = 0;
12
13              for( int x=0 ; x<5; x++) {
14                  Console.WriteLine($"Ingrese {x+1} numero: ");
15                  num[x] = int.Parse(Console.ReadLine());
16                  sum += num[x];
17              }
18              Console.WriteLine($"La Suma es: {sum}");
19          }
20      }
21  }
```

Solicita número al usuario

Acumula contenido de arreglo



```
C:\WINDOWS\system32\cmd.exe
Ingrese 1 numero:
1
Ingrese 2 numero:
2
Ingrese 3 numero:
3
Ingrese 4 numero:
4
Ingrese 5 numero:
5
La Suma es: 15
Presione una tecla para continuar . . .
```

# ARREGLOS DE UNA DIMENSIÓN



Programe un arreglo con 5 elementos de números aleatorios de 2 dígitos y visualícelos:

Para aleatorios entre los números MIN y MAX:

```
Random rand = new Random();  
int alea = rand.Next(MÍN, MÁX+1);
```

```
Random rnd = new Random();  
int month = rnd.Next(1, 13); // creates a number between 1 and 12  
int dice = rnd.Next(1, 7); // creates a number between 1 and 6  
int card = rnd.Next(52); // creates a number between 0 and 51
```

# ARREGLOS DE UNA DIMENSIÓN



Puede crear la cantidad de elementos según especifique el usuario.

Por ejemplo: Llene un arreglo con n elementos de números aleatorios de 2 dígitos y visualícelos:

```
1 using System;
2 using System.Collections.Generic;
3 using System.Linq;
4 using System.Text;
5 using System.Threading.Tasks;
6
7 namespace AppClase11 {
8     class Program {
9         static void Main(string[] args) {
10             Console.WriteLine("Ingrese cantidad de elementos:");
11             int n = int.Parse(Console.ReadLine());
12             int[] numeros = new int[n];
13             Random ran = new Random();
14
15             for( int x=0 ; x<n; x++) {
16                 numeros[x] = ran.Next(10, 99 + 1);
17             }
18             string todos = String.Join(", ", numeros);
19             Console.WriteLine($"numeros: {todos}");
20         }
21     }
22 }
```

Se crean tantos elementos como indica el usuario

Llena arreglo con aleatorios

Concatena los elementos del arreglo



```
C:\WINDOWS\system32\cmd.exe
Ingrese cantidad de elementos:
6
numeros: 47, 51, 52, 46, 99, 20
Presione una tecla para continuar
```



# ARREGLOS DE UNA DIMENSIÓN



```
1 using System;
2 using System.Collections.Generic;
3 using System.Linq;
4 using System.Text;
5 using System.Threading.Tasks;
6
7 namespace AppClase11 {
8     class Program {
9         static void Main(string[] args) {
10             int[] num1 = new int[] { 5, 4, 3, 2, 1 };
11             int[] num2 = { 10, 20, 30, 40, 50 };
12
13             Console.WriteLine($"Elementos num1: {num1.Length}");
14             Console.WriteLine($"Suma num2: {num2.Sum()}");
15             Console.WriteLine($"Promedio num2: {num2.Average()}");
16             Console.WriteLine($"Minimo num1: {num1.Min()}");
17             Console.WriteLine($"Maximo num1: {num1.Max()}");
18
19             Array.Sort(num1);
20             Console.WriteLine($"{{String.Join(" ", num1)}}");
21
22             Array.Reverse(num2);
23             foreach (int ele in num2) {
24                 Console.Write($"{{ele}}, ");
25             }
26         }
27     }
28 }
```

Se puede llenar indicando directamente los elementos

Tiene métodos para saber la cantidad de elementos, la suma, promedio, máximo, etc.

Ordena arreglo y lo muestra

Invierte la posición de elementos del arreglo y lo muestra



```
OWS\system32\cmd.exe
Elementos num1: 5
Suma num2: 150
Promedio num2: 30
Minimo num1: 1
Maximo num1: 5
1 2 3 4 5
```

# ARREGLOS DE UNA DIMENSIÓN



Mediante el for se puede buscar un elemento por comparaciones sucesivas. A continuación un ejemplo de búsqueda del mínimo, máximo y de un determinado valor:

- **Búsqueda del menor**

```
menor = a[0];  
for (i=1;i<n;i++)  
    if ( a[i]<menor )  
        menor=a[i];
```

- **Búsqueda del mayor**

```
mayor= a[n-1];  
for (i=0;i<n-1;i++)  
    if ( a[i]>mayor )  
        mayor=a[i];
```

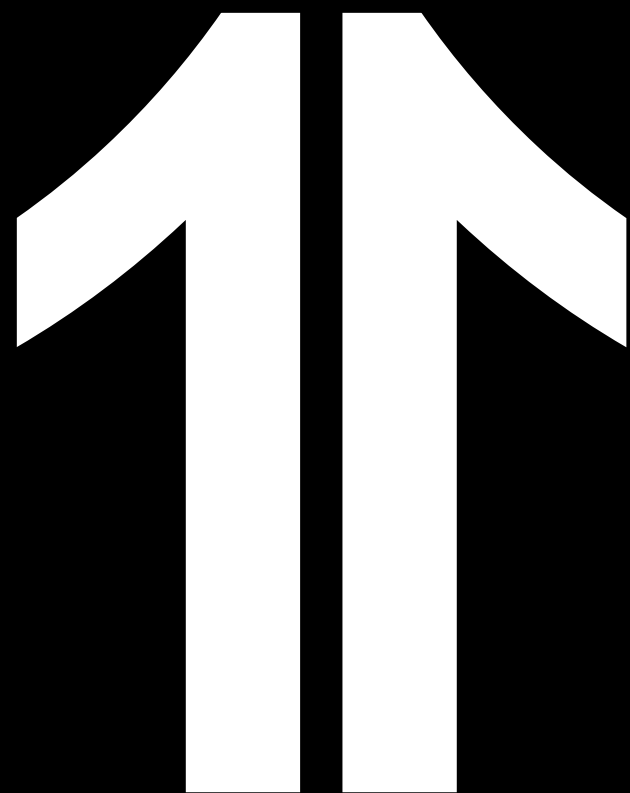
- **Búsqueda de elemento**

```
encontrado=-1;  
for (i=0;i<n;i++)  
    if ( a[i]==elemento_buscado )  
        encontrado=i;
```



## ¿PREGUNTAS?





**UPN**

**UNIVERSIDAD  
PRIVADA  
DEL NORTE**