

# INSTITUTO TECNOLÓGICO DEL SUR DE NAYARIT



ALUMNO:

Javier Eduardo Ramirez Ornelas

MAESTRA:

Cinthia Anahí Mata Bravo

NUMERO DE CONTROL:

191140026

MATERIA:

Programación orientada a objetos

CARRERA:

ITIC

## índice

<b>¿Qué es un arreglo?</b>	<b>3</b>
Arreglos unidimensionales	3
Arreglos multidimensionales	5
¿Cómo se declaran en C#?	6
Como Declarar e Inicializar un Arreglo en C#	6
Declaración de un Arreglo de tipo Vector en C#	6
Arreglos Multi-dimensionales	6
Arreglo de objetos	7
¿Cómo se declaran los arreglos de objetos de clases en c#?	8
Referencias bibliográficas:	11
Conclusión:	12

## ¿Qué es un arreglo?

se encuentran ubicados en forma consecutiva en la memoria RAM (sirve para almacenar datos en forma temporal).

- Tipos de arreglo
  - Arreglos unidimensionales.
  - Arreglos multidimensionales.

## Arreglos unidimensionales

Es un tipo de datos estructurado que está formado de una colección finita y ordenada de datos del mismo tipo. Es la estructura natural para modelar listas de elementos iguales. Están formados por un conjunto de elementos de un mismo tipo de datos que se almacenan bajo un mismo nombre, y se diferencian por la posición que tiene cada elemento dentro del arreglo de datos. Al declarar un arreglo, se debe inicializar sus elementos antes de utilizarlos. Para declarar un arreglo tiene que indicar su tipo, un nombre único y la cantidad de elementos que va a contener.

Un arreglo se define como una colección finita, homogénea y ordenada de elementos como se ha definido antes, pero se caracteriza por ser unidimensional porque tiene una sola dimensión como se muestra a continuación.

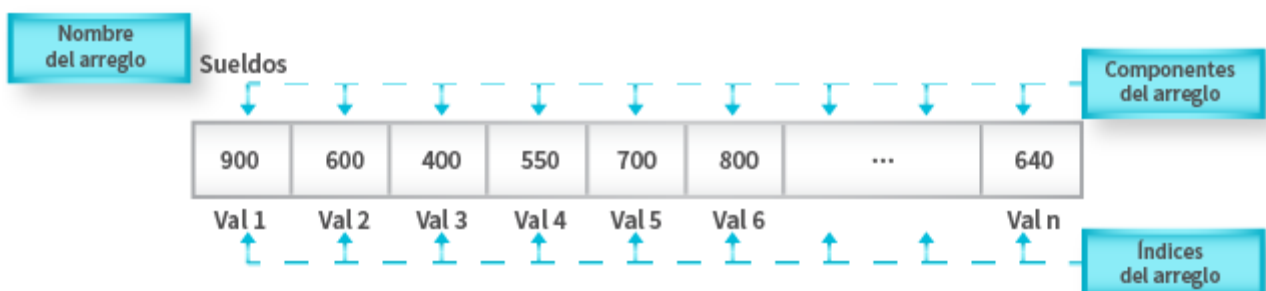
Un arreglo tiene la característica de que puede almacenar a N elementos del mismo tipo y además permite el acceso a cada uno de estos elementos. Se distinguen dos partes en los arreglos:

- Los componentes. Hacen referencia a los elementos que componen o forman el arreglo. Es decir, son los valores que se almacenan en cada una de sus casillas.
- Los índices. Son los que permiten acceder a los componentes del arreglo en forma individual. Es decir, la posición dentro del arreglo.

Para hacer referencia a un componente de un arreglo se necesita:

- El nombre del arreglo
- El índice del elemento

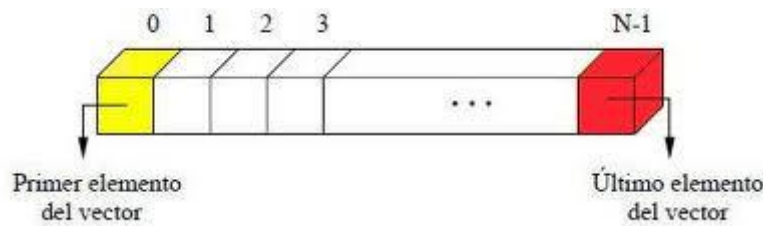
En la siguiente figura se presenta el ejemplo de un arreglo y se indican sus componentes y sus índices.



Un arreglo puede representarse gráficamente como se muestra en la figura:



## Arreglos multidimensionales



Con esta imagen es posible observar que un arreglo unidimensional es una estructura, dividida en celdas, donde cada una de ellas representa un espacio de almacenamiento. Para acceder a sus posiciones se utilizan los pequeños números que en la figura aparecen la parte superior, se les denomina índices, y representan las posiciones. El primer índice en realidad depende del software, y generalmente es 0 o 1. En el caso del lenguaje de programación C y Java, este índice inicial es 0.

La imagen también permite demostrar que es FINITO; así es posible observar que tiene un inicio, en la posición 0 y un final en la posición N-1, donde N es el tamaño del arreglo. Es importante recordar que el tamaño del arreglo, representa la cantidad de memoria que se reservará para usar esta estructura, por lo que NO podrá variarse durante el tiempo de ejecución de un programa.

Este arreglo deberá tener un nombre, conservando las recomendaciones para asignar el nombre a un identificador, es decir que empiece con una letra, que no tenga espacios ni caracteres especiales, que utilice de preferencia si es necesario el convenio Camellize, que sea claro y corresponda con la información que va a contener. Así, por ejemplo, si va a almacenar las notas de un grupo de estudiantes podría llamar notas o notas Programación.

## ¿Cómo se declaran en C#?

Los arreglos en C# (también conocidos como Arrays) al igual que en C/C++, son indexados iniciando en cero (0). La forma de trabajo es muy similar a la mayoría de lenguajes, pero hay lagunas diferencias que notarán.

## Como Declarar e Inicializar un Arreglo en C#

Cuando vayamos a declarar un arreglo en debemos colocar los corchetes después de tipo de dato. En C/C++ se estila poner los corchetes después del identificador, eso causa un error de compilación en C#-

## Declaración de un Arreglo de tipo Vector en C#

```
//Declaración errónea
int valores [];
//Declaración valida
int[] valores;
```

En C# podemos indicar el tamaño del arreglo luego de la declaración. Esto nos permite decidir qué tamaño va a tener el arreglo según lo que necesitemos.

```
int [] valores; //valores sin inicializar
valores = new int [100]; //100 elementos
valores = new int [20]; //ahora contiene 20 elementos
```

## Arreglos Multi-dimensionales

En C# también podemos declarar arreglos multidimensionales, aquí unos ejemplos:

```
//Arreglos unidimensionales o de dimensión simple
int[] valores1; //sin inicializar
int[] valores2 = new int[50];

//Arreglos multidimensionales
int[,] valores1; //sin inicializar
int[,] valores2 = new int[3,7];
int[,,,] valores3 = new int[3,4,2]; //Arreglo de tres dimensiones

//Arreglo de arreglos
int[][] matriz; //sin inicializar
//Los arreglos de arreglos se inicializan de manera diferente
int[][] matriz = new int[3][];
for (int i = 0; i < matriz.Length; i++)
{
    matriz[i] = new int[4];
}
```

## **Arreglo de objetos.**

La función básica de un arreglo es almacenar en una variable más de un valor de un mismo

tipo de dato, por ejemplo, la siguiente declaración `int[ ] numero= new int [5];` permite almacenar

en la variable número, 5 valores enteros.

En las clases el concepto de arreglos es el mismo, con la diferencia que ahora se almacenarán

objetos de una clase o de diferentes clases.

Los objetos se pueden estructurar como un array. Los objetos son variables y tienen las

mismas capacidades y atributos que cualquier tipo de variables, por tanto es posible disponer

objetos en un array.

La sintaxis es exactamente igual a la utilizada para declarar y acceder al array.

También disponemos de arrays bidimensionales.

Cuando se crea un array de objetos éstos se inicializan llamando al constructor sin argumentos. Por consiguiente, siempre que se prevea organizar los objetos en un array, la clase debe tener un constructor que pueda llamarse sin parámetros.

## ¿Cómo se declaran los arreglos de objetos de clases en c#?

```
using System;

public class Ejemplo_07_02b
{
    public static void Main()
    {
        Animal[] misAnimales = new Animal[8];

        misAnimales[0] = new Perro();
        misAnimales[1] = new Gato();
        misAnimales[2] = new GatoSiames();

        for (byte i=3; i<7; i++)
            misAnimales[i] = new Perro();

        misAnimales[7] = new Animal();
    }
}

// -----

public class Animal
{
    public Animal()
    {
        Console.WriteLine();
        Console.WriteLine("Ha nacido un animal");
    }
}

// -----

public class Perro: Animal
{
    public Perro()
```



```

        {
            Console.WriteLine("Ha nacido un perro");
        }
    }

// -----

public class Gato: Animal
{
    public Gato()
    {
        Console.WriteLine("Ha nacido un gato");
    }
}

// -----

public class GatoSiames: Gato
{
    public GatoSiames()
    {
        Console.WriteLine("Ha nacido un gato siamés");
    }
}

```

La salida de este programa sería:

Ha nacido un animal

Ha nacido un perro

Ha nacido un animal

Ha nacido un gato

Ha nacido un animal

Ha nacido un gato

Ha nacido un gato siamés

Ha nacido un animal

Ha nacido un perro

Ha nacido un animal

Ha nacido un perro

Ha nacido un animal

Ha nacido un perro

Ha nacido un animal

Ha nacido un perro

Ha nacido un animal

## **Referencias bibliográficas:**

<http://www.nachocabanes.com/csharp/curso2015/csharp07b.php>

<http://www.udb.edu.sv/udb/archivo/guia/informatica-ingenieria/programacion-ii/2016/i/guia-6.pdf>

[https://www.ecured.cu/Arreglos\\_\(Inform%C3%A1tica\)#Arreglos\\_multidimensionales](https://www.ecured.cu/Arreglos_(Inform%C3%A1tica)#Arreglos_multidimensionales)

<http://www3.uacj.mx/CGTI/CDTE/JPM/Documents/IIT/arreglos1/unidimensional.html>

<http://csharp-facilito.blogspot.com/2013/07/arreglos-en-c-sharp.html>

## Conclusión:

Un arreglo es una colección de variables relacionadas a las que se hace referencia por medio de un nombre común.

Lo que entendí del arreglo unidimensional es como algo que te está guardando todos los datos que introduce, pero dentro de corchetes debe venir un número o valor de las variables que quieras guardar es decir quieres guardar el sueldo de un trabajador y pones tu (n en 7) es decir quieres que por 7 semanas te calcule un sueldo total y lo que va haciendo las va guardando hasta que llegue a ese número de sueldos guardados vi que debía llevar un índice es algo que no entendí también que tu un arreglo tiene que indicar su tipo, un nombre único y la cantidad de elementos que va a contener.

En relación a este tema siento que es como lo que vimos en fundamentos de programación, pero más avanzado es decir antes poníamos la variable num1 y dentro de un if poníamos para que se retornara ciertas veces hasta que llegara hay y te arrojará el resultado que tu querías.

Los arreglos multidimensionales se observan claramente en forma de matrices, de las cuales podemos hacer operaciones fundamentales como: suma, resta y multiplicación.

Los arreglos multidimensionales es casi lo mismo que los unidimensionales

Sirven para guardar información su índice siempre comienza en 0 o 1 para llevar un conteo de las cosas que introduces vi algo diferente y es que termina en n-1 y N es la memoria que se va a ocupar lo de n-1 es como para decirle al programa un resultado o como un stop.

Debes declararlo con un nombre específico es decir como si vas a guardar el sueldo de todos tus trabajadores que sea algo claro como sueldo o sueldos.

No es algo difícil de comprender, pero si de crear es algo revoltoso, pero lo tratare de hacer.

Se me hace algo complicado porque pues solo estoy aprendiendo mediante videos o investigando información y no mediante practica esta unidad.

me la he pasado un poco tenso porque justo cuando ya le estaba entendiendo pasa esto y es algo que me estresa demasiado por que no se hacer las cosas y cuando estábamos en el salón pues tenía a lado a neto o a Jaqueline que me decían mira esto se hace así o estas mal aquí el orden de como acomodar mis botones se me dificulta bastante y vamos avanzando y viendo cosas nuevas es algo enfadoso porque me gusta la programación pero me estoy atrasando y no estoy entendiendo los temas que vienen pero voy a tratar de ver muchos videos para comprender mejor los temas creo que mi conclusión seria toda no entendí mucho de este tema.