

UPN

**UNIVERSIDAD
PRIVADA
DEL NORTE**



CADENAS DE CARACTERES

SEMANA 14

AGENDA



III	Arreglos 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 Evaluación T3
		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	EVALUACIÓN FINAL



Al término de la clase, el estudiante elabora programas haciendo uso del lenguaje C# y el Visual Studio, configurando las funciones con cadenas de caracteres, mostrando una lógica coherente al implementar su algoritmo



CADENAS DE CARACTERES

El lenguaje de programación C# tiene algunas funciones para realizar operaciones con cadenas de caracteres (string), que nos serán de gran utilidad.



FUNCIÓN	DESCRIPCIÓN
Length	Permite obtener la longitud de la cadena.
SubString	Recupera una subcadena indicándole la ubicación inicial y la cadena de caracteres.
IndexOf	Devuelve el índice de un carácter específico.
ToUpper	Convierte a mayúsculas los caracteres de la cadena.
ToLower	Convierte a minúsculas los caracteres de la cadena.
Replace	Reemplaza un carácter o caracteres de una cadena por otros caracteres.

DESCUBRIENDO



```
using System;
namespace Semana13
{
    0 referencias
    class Program
    {
        0 referencias
        static void Main(string[] args)
        {
            Console.Write("Ingrese una cadena de texto: ");
            string cadena = Console.ReadLine();
            //Función Length
            int longitud = cadena.Length;
            Console.WriteLine("La cadena tiene una longitud de {0} caracteres", longitud);

            //Funcion SubString
            string subcadena = cadena.Substring(1, 4);
            Console.WriteLine("La subcadena es: {0}", subcadena);

            //Función IndexOf
            int index = cadena.IndexOf("A");
            Console.WriteLine("La posición de la letra A es: {0}", index);

            //Funcion ToUpper
            string mayusculas = cadena.ToUpper();
            Console.WriteLine("El texto en mayusculas es: {0}", mayusculas);

            //Funcion ToLower
            string minusculas = cadena.ToLower();
            Console.WriteLine("El texto en minusculas es: {0}", minusculas);

            //Función Replace
            string reemplaza = cadena.Replace(" ", "");
            Console.WriteLine("Texto sin espacios en blanco: {0}", reemplaza);
            Console.ReadKey();
        }
    }
}
```



Dividir una cadena en subcadenas (al igual que dividir una frase en palabras) es una tarea común de programación. El método `Split()` toma una matriz de `char` de delimitadores (por ejemplo, un carácter de espacio) y devuelve una matriz de subcadenas. Para obtener acceso a esta matriz con `foreach`:



```
using System;
namespace Semana13
{
    0 referencias
    class Program
    {
        0 referencias
        static void Main(string[] args)
        {
            char[] delimit = new char[] { ' ' };

            Console.Write("Ingrese una cadena de texto: ");
            string cadena = Console.ReadLine();

            foreach (string substr in cadena.Split(delimit))
            {
                Console.WriteLine(substr);
            }
            Console.ReadKey();
        }
    }
}
```



Los objetos String también tienen un método `CompareTo()` que devuelve un valor entero, basado en si una cadena es menor que ($<$) o mayor que ($>$) otra. Al comparar las cadenas, se utiliza el valor Unicode, y las minúsculas tienen un valor menor que las mayúsculas.



```
using System;
namespace Semana13
{
    0 referencias
    class Program
    {
        0 referencias
        static void Main(string[] args)
        {
            String cadena1 = "FUNDAMENTOS DE ALGORITMOS";
            String cadena2 = "fundamentos de algoritmos";

            int numero = cadena1.CompareTo(cadena2);
            Console.WriteLine("El valor de numero es: {0}", numero);
            Console.ReadKey();
        }
    }
}
```



Los objetos String también tienen un método `Contains()` que devuelve un `true` (Verdadero) si la cadena contiene los caracteres que se especifica, en caso contrario devuelve `false` (Falso);

DESCUBRIENDO



```
Program.cs*  + X
c# Semana13  Semana13.Program

1  using System;
2  namespace Semana13
3  {
4      0 referencias
5      class Program
6      {
7          0 referencias
8          static void Main(string[] args)
9          {
10             String cadena = "Fundamentos de Algoritmos";
11             Console.WriteLine(cadena.Contains("Algo"));
12         }
13     }
14 }
```

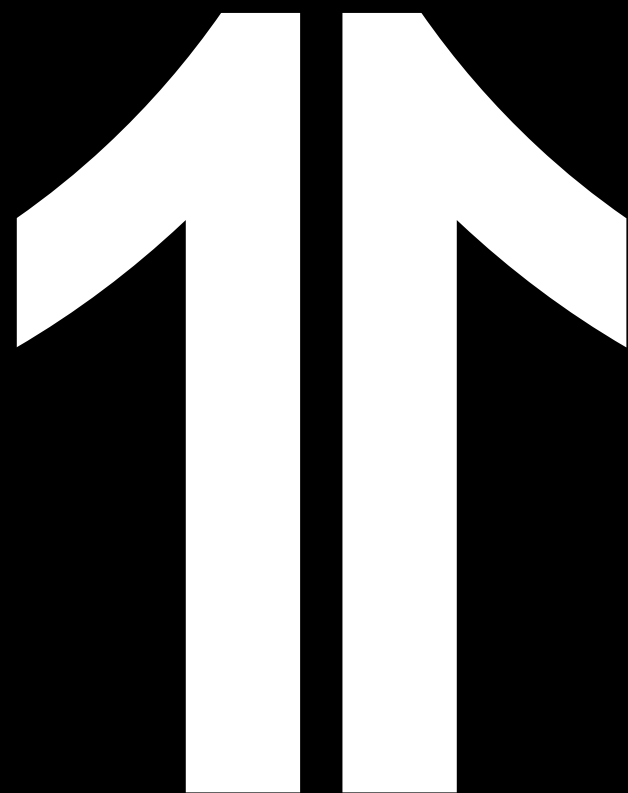


Desarrolle los ejercicios propuestos en nuestra aula virtual.



¿PREGUNTAS?





UPN

**UNIVERSIDAD
PRIVADA
DEL NORTE**