

Universidad Rafael Landívar  
Facultad de Ingeniería.  
Ingeniería Mecánica Industrial  
Laboratorio de Introducción a la programación -  
Sección: Sección 09  
Catedrático: Ing. Herwing Alexis Rodríguez Franco  
Alumno-Asistente: Diego Villanada



## **LABORATORIO 4**

### **Estructura de un Programa y Operaciones Aritméticas**

Estudiante: Aldo Josué Guevara Catalán  
Carné: 1326819

Guatemala, 20 de septiembre de 2023

## **Definición de conceptos**

### **a. Definición e inicialización de variables**

La definición e inicialización de variables es un proceso fundamental en la programación. Consiste en declarar una variable (especificar su tipo de datos) y asignarle un valor inicial. Esto permite reservar un espacio en la memoria para la variable y almacenar un valor en ella. Por ejemplo, en muchos lenguajes de programación, puedes definir e inicializar una variable de entero de la siguiente manera:

```
Int edad = 25;
```

### **b. Diferencias entre tipos de datos numéricos, booleanos y cadenas**

Tipos de datos numéricos en C#: Representan valores numéricos, como enteros (int) o números de punto flotante (float o double). Se utilizan para realizar cálculos matemáticos. Ejemplos

```
Int numero = 5;  
Double pi = 3.14;
```

Tipos de datos booleanos en C#: Representan valores de verdad, es decir, true (verdadero) o false (falso). Se usan principalmente en expresiones lógicas y condicionales. Ejemplos:

```
bool es Verdadero = true;  
bool es Falso = false;
```

Tipos de datos de cadenas en C#: Representan secuencias de caracteres, como texto. Se utilizan para almacenar y manipular texto. Ejemplo:

```
string mensaje = "Hola, mundo";
```

### **c. Operaciones Aritméticas (incluyendo DIV y MOD)**

Las operaciones aritméticas en C# son similares a las operaciones en otros lenguajes y se realizan de la siguiente manera:

```
int x = 10;  
int y = 3;  
int suma = x + y; // suma contiene 13  
int resta = x - y; // resta contiene 7  
int multiplicación = x * y; // multiplicación contiene 30  
int división = x / y; // división contiene 3 (la división de enteros trunca el resultado)  
int modulo = x % y; // modulo contiene 1 (el residuo de la división)
```

#### **d. Diferencias entre asignar (operador =) un valor a una variable y comparaciones de igualdad (operador ==)**

El operador = se utiliza para asignar un valor a una variable en C#. Por ejemplo,

```
x = 10; asigna el valor 10 a la variable x.
```

El operador == se utiliza para comparar dos valores y verificar si son iguales en C#. Por ejemplo,

```
x == 10; verifica si el valor de x es igual a 10 y devuelve true o false según el resultado de la comparación.
```

#### **e. Jerarquía de operadores**

La jerarquía de operadores en C# sigue las reglas matemáticas estándar, como PEMDAS (Paréntesis, Exponentes, Multiplicación y División, Suma y Resta), al igual que en otros lenguajes.

#### **f. Comentarios dentro del código**

Los comentarios en C# se crean utilizando // para comentarios de una sola línea y /\* \*/ para comentarios de varias líneas:

```
// Esto es un comentario de una sola línea en C#  
  
/*  
Esto es un comentario  
de varias líneas en C#  
*/
```

#### **g. Input/output de datos**

En C#, puedes utilizar la clase Console para realizar entrada y salida de datos en la consola. Por ejemplo:

```
string nombre = Console.ReadLine(); // Obtener entrada del usuario  
Console.WriteLine("Hola, " + nombre + "! Bienvenido."); // Mostrar resultados en la consola
```

#### **h. Conversión entre tipos de datos**

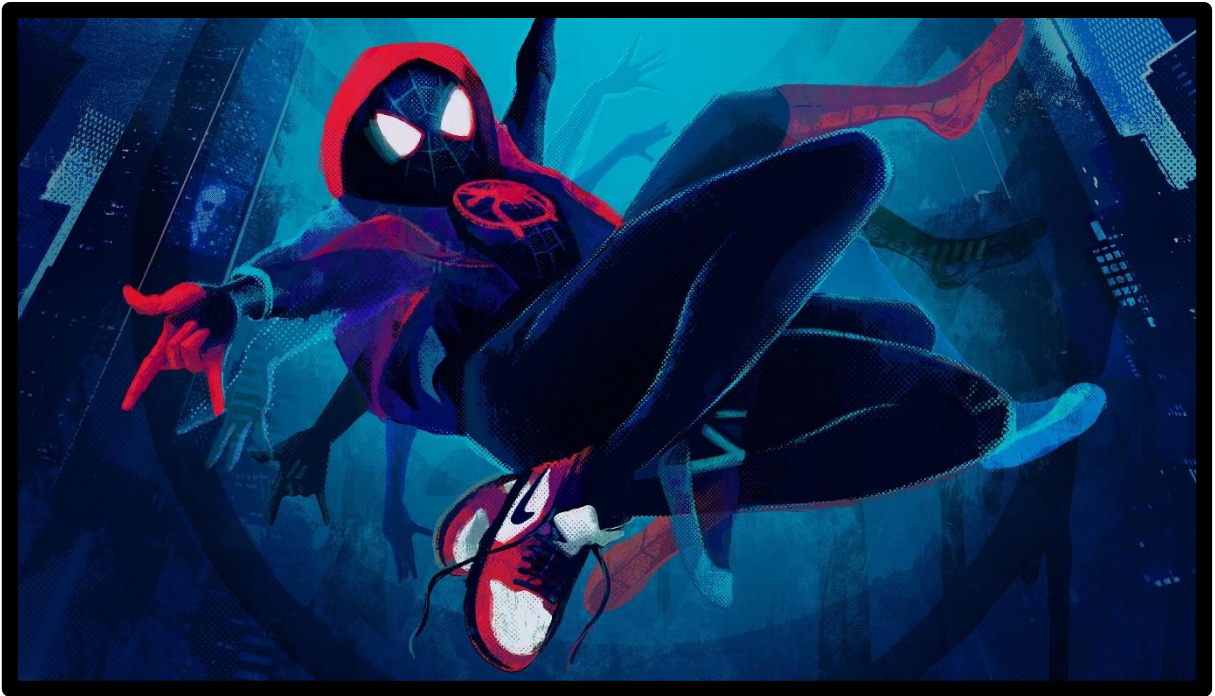
Para realizar conversiones entre tipos de datos en C#, puedes utilizar operadores de conversión o métodos específicos de cada tipo. Por ejemplo:

```
int numero = 42;  
string cadena = numero.ToString(); // Convierte el número a una cadena
```

#### **i. Concatenación básica de cadenas**

La concatenación de cadenas en C# se puede hacer utilizando el operador + para combinar cadenas:

```
string nombre = "Juan";  
string apellido = "Pérez";  
string nombre Completo = nombre + " " + apellido; // Concatena las dos cadenas  
con un espacio en medio
```



*Nunca sabrás cuando estarás listo, tendrás que dar un salto de fe*  
*-Spiderman-Miles Morales*