

Tema 2: Tipos de Datos

1. Introducción

Los tipos de datos permiten almacenar diferentes clases de información en las variables. Son fundamentales para controlar el comportamiento y el uso de memoria en un programa.

2. Tipos de datos primitivos

En C++, los tipos más comunes incluyen:

- **int**: Números enteros
- **float**: Números decimales de precisión simple
- **double**: Números decimales de doble precisión
- **char**: Caracteres individuales
- **bool**: Valores booleanos (true/false)

Tipo	Descripción	Tamaño típico	Rango aproximado
int	Número entero	4 bytes	-2,147,483,648 a 2,147,483,647
float	Número real (precisión simple)	4 bytes	$\pm 3.4e\pm 38$ (~7 cifras decimales)
double	Número real (precisión doble)	8 bytes	$\pm 1.7e\pm 308$ (~15 cifras decimales)
char	Carácter ASCII	1 byte	0 a 255 (con unsigned)
bool	Lógico	1 byte	true o false

3. Modificadores de tipo

C++ permite modificar los tipos básicos con palabras como: short, long, unsigned, signed.

Ejemplo: unsigned int, long long int

4. Declaración y uso

```
int edad = 17;  
float promedio = 8.5;  
double pi = 3.14159265;  
char letra = 'A';  
bool aprobado = true;
```

5. Conversión de tipos (casting)

La conversión puede ser automática (implícita) o manual (explícita):

```
int a = 5;
double b = a;           // conversión implícita
double x = 5.7;
int y = (int)x;        // conversión explícita: y = 5
```

6. Buenas prácticas

- Usa long long para números muy grandes.
- Prefiere double sobre float cuando se necesita más precisión.
- Inicializa siempre las variables.
- Usa nombres descriptivos.

7. Ejemplos aplicados

Ejemplo 1: Verificar si un número es par usando bool

```
int n;
cin >> n;
bool esPar = (n % 2 == 0);
cout << boolalpha << esPar << endl;
```

Ejemplo 2: Leer tres números decimales y calcular el promedio

```
double a, b, c;
cin >> a >> b >> c;
double promedio = (a + b + c) / 3;
cout << fixed << setprecision(2) << promedio << endl;
```

Ejemplo 3: Mostrar el código ASCII de un carácter

```
char c;
cin >> c;
cout << "ASCII: " << (int)c << endl;
```

8. Ejercicios de práctica

1. Declara variables de tipo int, float, double, char y bool y muestra sus valores.
2. Lee dos enteros y muestra su producto. Usa long long si esperas un número muy grande.
3. Lee un número real y muéstralo con 3 decimales.
4. Lee un carácter y muestra su valor ASCII.
5. Lee una edad (int) y muestra true si es mayor o igual a 18, false en caso contrario.

6. Lee un double y conviértelo en entero con cast. Muestra ambos.
7. Calcula el promedio de 3 números reales.
8. Lee dos enteros grandes y muestra su suma usando long long.
9. Declara un bool llamado esPar que sea true si un número entero es par.
10. Declara una variable char y úsala para mostrar un mensaje personalizado.