



Objetivo de la práctica:

- Conocer los límites de representación de los tipos de datos básicos
- Realizar programas de cálculo sencillo utilizando operadores aritméticos
- Utilizar funciones básicas de entrada / salida.

CONCEPTOS BÁSICOS

Los tipos básicos y sus límites de representación

Los tipos de datos básicos que se van a manejar en esta práctica son:

- Carácter: **char** (-128 a 127)
- Entero: **short** (-32763 al 32762), **long**(de -2147483648 a 2147483647)
- Real: **float** 3.4E-38 y 3.4E+38 , **double** 1.7E-308 y 1.7E+308.

Es importante fijarse en los rangos de representación ya que pueden inducir a errores de funcionamiento difíciles de detectar.

Operadores aritméticos.

Los operadores aritméticos se muestran en la tabla con su orden de precedencia:

Mayor prioridad	↑	++ --
		-(menos monario)
		* / %
		+ -
Menor	↓	=, +=, -=, *=, /=, %=

Es necesario tener esa prioridad en cuenta a la hora de escribir expresiones aritméticas. Si se desea cambiar la precedencia y forzar operaciones se deben utilizar paréntesis.

Otro elemento importante a recordar es que los operadores trabajan de distinta forma en función de los tipos de datos a los que se enfrentan. Es necesario tener esto muy en cuenta ya que puede inducir a errores. Por ejemplo la división de dos enteros siempre será un entero. Para evitar estos problemas podemos recurrir a lo que se denomina conversión de tipos o *casting*. Este se realiza poniendo delante del dato a cambiar y entre paréntesis el tipo al cual queremos convertirlo: (float)9

Funciones de Entrada Salida (asociadas a iostream)

La función de salida básica a emplear será: **cout**, junto con el operador inserción <<

```
cout << "El precio total es: << (precio1+precio2);
```

A esto se pueden añadir secuencias de escape:

Carácter	acción
\n	Salto de línea (equivale a endl)
\t	Tabulador
\\	Barra Invertida
\"	Comillas dobles.



Se pueden controlar ciertas características de la salida con modificadores sobre cout:

Modificador de formato	Resultado
cout.precision(Nº Cifras)	La precisión decimal se limita a las cifras indicadas
cout.width(Nº caracteres)	Controla el tamaño mínimo de la salida
cout.fill(caracter)	Indica el carácter con el que se completa una salida que no llega al mínimo de longitud indicado en width.

La función de entrada será: **cin** con el operador inserción en la otra dirección >>
cin >> num_empleados;

BLOQUE DE EJERCICIOS

1. Prueba el siguiente programa; si te parece que no funciona correctamente modifícalo hasta que funcione como debería.

```
#include <iostream>
using namespace std;
int main()
{
    short c1=20;
    short c2=5000;
    short c3;
        c3 = c1 * c2;
        cout << c1 << " X " << c2 << " = " << c3 << endl;
        system("pause");
        return 0;
}
```

2. Prueba el siguiente programa; si te parece que no funciona correctamente modifícalo hasta que funcione como debería.

```
#include <iostream>
using namespace std;
int main()
{
    float c1=0.1000000005;
    float c2=0.1000000007;
        if(c1==c2)
            cout << "c1 y c2 son iguales" << endl;
        else
            cout << "c1 y c2 son distintos" << endl;

        system("pause");
        return 0;
}
```



3. Prueba el siguiente programa; si te parece que no funciona correctamente modifícalo hasta que funcione como debería.

```
#include <iostream>
using namespace std;
int main()
{
    int c1 = 3;
    int c2 = 2;
    flota c3;

    c3 = c1 / c2;
    cout << c1 <<"entre " <<c2<<"=" <<c3<<endl;
    return 0;
}
```

4. Escribe un programa en el cual se introduzca como entrada una longitud en Kilómetros y nos devuelva como resultado su equivalente en millas. (1 milla=1.609 Km).
5. Según el algoritmo de Leibniz, es posible hallar el valor de π a partir de la siguiente serie:
$$\pi = 4(1/1-1/3+1/5-1/7+1/9\dots)$$

Escribe un programa que calcule una aproximación de ese valor utilizando tan sólo 7 elementos de la serie. Comprueba que el resultado de la aproximación es 3.28374
6. Realiza un programa que lea un número enteros de 3 cifras y determine la suma de sus cifras.
7. Escribir un programa que recoja por teclado dos números reales y calcule su suma mostrándola por pantalla con dos cifras decimales.
8. Escribe un programa que lea 5 valores reales y los saque por pantalla organizados de la siguiente forma:
###65.40
#2345.56
####4.78
....
9. Realiza un programa que calcule la hora que será transcurridos una determinada cantidad de segundos. El programa pedirá al usuario la hora actual (hora, minuto y segundos) y el intervalo de tiempo medido en segundos. El programa generará como resultado la hora que será cuando transcurra ese intervalo de tiempo.