

# Algoritmos y Estructuras de Datos

Oscar Meza

[oscarmeza@gmail.com](mailto:oscarmeza@gmail.com)

Veamos si Visual Studio Code está funcionando....

(crear y abrir carpeta “pruebas” y luego crear archivo prueba.cpp)

Y ejecutemos el siguiente programa en C++:

```
#include <iostream>

int main() {
    std::cout << "    *    \n"
               << "   ***   \n"
               << "  ***** \n"
               << "    *    \n"
               << "    *    \n"
               << "    *    \n";
}
```

Lo podemos compilar y ejecutar en una terminal: >g++ -o prueba prueba.cpp

## 1) Crear archivo prueba1.cpp con el siguiente programa:

```
#include <iostream>
```

```
int main() {
```

```
    int xx = 5, yy = 3, int zz = 2;
```

```
    yy += xx++ + zz;
```

```
    char a = 'a';
```

```
    std::cout << yy << " " << xx << " " << a << " "
```

```
                << static_cast<int>(a) << '\n';
```

```
    std::cout <<(xx>yy) << " " << std::boolalpha << (xx>yy) << '\n';
```

```
}
```

probar (y modificar para ver comportamiento de las operaciones --, ++, \*=, -=, /=):

## 2) Cree nuevo archivo prueba2.cpp.

Hacer el siguiente programa que lee dos números e imprime su promedio negativo:

```
#include <iostream>
```

```
int main() {
```

```
    int value1, value2, sum;
```

```
    std::cout << "Coloque dos valores y teclee enter: ";
```

```
    std::cin >> value1 >> value2; // entrada por teclado, tecleamos por ejemplo: 2 3 enter
```

```
    sum = -(value1 + value2)*2;
```

```
    std::cout << "-( " << value1 << " + " << value2 << ") * " << 2 << " = " << sum << "\n";
```

```
}
```

### 3) Cree archivo prueba3.cpp

Hacer programa que lea 4 números enteros determine cual es el menor de los cuatro y lo imprima. **Utilice solamente el IF , no cree nuevas variables.**

### 4) Cree un archivo prueba4.cpp:

Hacer programa que lea un número entre 0 y 5 e imprima por pantalla el número pero en español. Si coloco 0 imprime “Colocó un cero”, etc.

### 5) Cree archivo prueba5.cpp

Hacer un programa que imprima todas las permutaciones de 0,1 2, 3, 4,5,6  
Utilice for(...)

6) Cree un archivo prueba6.cpp:

Hacer programa que lea un número entero no negativo n e imprima n!

7) Cree un archivo prueba7.cpp.

Dado un numero entero m y un numero entero positivo, en C++ la operación  $m \% n$  determina el resto de la división entera de m entre n:

$(-3) \% 2 = -1$  porque  $-3 = 2(-1) + (-1)$  resto negativo

$(-3) \% 2 = 1$  porque  $3 = 2 \times (-1) + (-1)$

$3 \% 2 = 1$  porque  $3 = 2 \times 1 + 1$

$3 \% (-2) = 1$  porque  $3 = (-2) \times (-1) + 1$

Sin embargo el teorema de la división euclidea dice que dados m entero y n entero entonces existen **dos únicos enteros c y r** con r **no negativo** y  $0 \leq r < |n|$  tal que  $m = n \times c + r$

Ejemplo:

$m = 10$  y  $n = 6$  tenemos  $c = 1$  y  $r = 4$

$m = 10$  y  $n = -6$  tenemos  $c = -1$  y  $r = 4$

$m = -10$  y  $n = 6$  tenemos  $c = -2$  y  $r = 2$

$m = -10$  y  $n = -6$  tenemos  $c = 2$  y  $r = 2$

Hacer un algoritmo iterativo que dados  $m$  y  $n$  enteros determine  $c$  y  $r$  según el teorema de la división euclidea. (No se puede utilizar %)

Haga también un algoritmo para hallar la división euclidea pero solo utilizando %