

# El lenguaje C

## 1. Ejemplos correspondientes al teórico2

### 1.1. Conceptos de memoria, nombres de variables

Ejemplos de nombres de variable validos son:

i, i1, i2, int1, int2, f, floa, double1

ejemplos de nombres de variable no válidos son:

2i, 4fl, int, float, double, if

los primeros son no válidos por empezar con números, los siguientes son palabras reservadas.

### 1.2. Declaración de constantes

ejemplo:

```
const float PI 3.1415926;
```

en general se utiliza:

```
const tipo nombre_de_variable valor;
```

lo que hacemos es declarar una variable inicializada con un valor. El declarar-la como const (poner const antes del tipo) es lo que hace que sea una constante. Las constantes no se pueden modificar (es un error de sintaxis).

### 1.3. Operaciones aritméticas

Ejemplos de usos de operaciones aritméticas en programas C son:

```
x=x*2+1;  
x=y*3-5+z;  
z=(x-1)/2;  
z=(x%2)*y;
```

donde utilizamos las operaciones dentro de expresiones que se encuentran en la operación de asignación.

## 1.4. Operadores de igualdad y relacionales

Ejemplos de operadores de igualdad y relacionales son los siguientes:

```
x+2 == y-1
(z*3)+1 != (y+2)*3
x>y
x+2 <= y+3
x-2 >= (y-1)*2
```

es usual utilizar expresiones de igualdad o relacionales en instrucciones específicas que contienen condiciones lógicas.

Si pensamos en pseudocódigo, las instrucciones si, si/si no y mientras contienen expresiones lógicas. Las instrucciones correspondientes en C también tienen condiciones lógicas y las escribimos como en los ejemplos de arriba.

## 1.5. Operadores booleanos

Ejemplos de operadores booleanos son condiciones lógicas como las anteriores conectadas por and, or y not. Por ejemplo:

```
(x > y && y >= z)
(x+2 == z && z+3<y)
(x || y); /* x e y son variables booleanas */
(x+3 < y%2 || !(z==y))
```

## 1.6. Operadores de asignación

Ejemplos de operadores de asignación:

```
c+=3*z; /* corresponde a c=c+3*z; */
d*=2-y+x; /* corresponde a d=d*(2-y+x); */
e%=2; /* corresponde a e=e%2; */
```

## 1.7. Operadores incrementales y decrementales

Podemos utilizarlos en el lugar de cualquier expresión aritmética, por ejemplo:

```
x=c++; /* asignacion */
y=x+(z++); /*asignacion */
x+2 == -- y; /* expresion booleana */
y*5 != ++z + --x; /* expresion booleana */
y-3 > ++x; /* expresion relacional */
y+x <= z -- * 5; /* expresion relacional */
```

## 1.8. Operador condicional

Ejemplos:

```
x>0 ? 5 : 3; /* retorna un valor */
```

```
x+y <= z*6 ? x=x+2 : x=x+1; /* ejecuta una instruccion */
```