



INSTITUTO  
**KHIPU**

**Semestre I**

**Sesión 19**

**CARRERA PROFESIONAL**

# **DESARROLLO DE SISTEMAS DE INFORMACIÓN**

**FUNDAMENTOS DE  
PROGRAMACIÓN**

**Tema**

**OPERADORES LOGICOS EN EXPRESIONES  
BOOLEANAS COMPUESTAS**



## CONCEPTOS BÁSICOS

Ahora hablaremos sobre los operadores lógicos AND(Y) y OR(O) en expresiones booleanas compuestas

### **Operador lógico AND (Y):**

El operador lógico AND se utiliza para combinar dos o más condiciones en una expresión booleana. La expresión resultante es verdadera únicamente si todas las condiciones individuales son verdaderas.

### **Operador lógico OR (O):**

El operador lógico OR se utiliza para combinar dos o más condiciones en una expresión booleana. La expresión resultante es verdadera si al menos una de las condiciones individuales es verdadera.

### **Asignaciones booleanas(Logicas)**

Las sentencias de asignación se pueden escribir de modo que se puede dar un valor de tipo bool o una variable bool.

### **Ejemplo**

MayorDeEdad = true;	Asigna el valor true MayorDeEdad
MayorDeEdad = (x == y);	Asigna el valor de x == y a MayorDeEdad
MayorDeEdad	Cuando x e y son iguales, MayorDeEdad es true y si no false.

Las sentencias de asignación siguientes asignan valores a los dos tipos de variables bool, rango y es\_letra. La variable rango es verdadera (true) si el valor de n está en el rango –100 a 100; la variable es\_letra es verdadera si car es una letra mayúscula o minúscula

```
rango = (n > -100) && (n < 100);
```

```
esletra = (('A' <= car) && (car <= 'Z')) ||
```

```
(( 'a' <= car) && (car <= 'Z'));
```

La expresión de a es true si n cumple las condiciones expresadas (n mayor de –100 y menor de 100); en caso contrario es false. La expresión b utiliza los operadores && y ||. La primera subexpresión (antes de ||) es true si car es una letra mayúscula; la segunda subexpresión (después de ||) es true si car es una letra minúscula. En resumen, esLetra es verdadera (true) si car es una letra, y false en caso contrario.

### **Expresiones booleanas**

La mayoría de las sentencias de bifurcación que se verán en los Capítulos 4 y 5 están controladas por expresiones booleanas o lógicas. Una expresión booleana es cualquier expresión que es verdadera o falsa. La forma más simple de una expresión booleana consta de dos expresiones, tales como números o variables que se comparan con uno de los operadores de comparación o relacionales. Observe que algunos de los operadores se escriben con dos símbolos, ==, !=, <=, >=, etc. y también, que se utiliza un signo doble igual para el signo igual y que se utilizan dos símbolos != para el signo no igual (Tabla 3.7).

### **Operador "and" &&**

Se pueden combinar dos comparaciones utilizando el operador "and" que se escribe &&.

### **Ejemplo**

La expresión (2 < x)&&(x < 7) es verdad si x es mayor que 2 y x es menor que 7, en caso contrario es falsa

```
(exp_booleana_1) && (expresión_booleana_2)
```

### **Aplicación**

```
if (nota > 0) && (nota < 10)
    cout << "la calificación está entre 0 y 10.\n";
else
    cout << "la calificación no está entre 0 y 10.\n";
```



si el valor de nota es mayor que 0 y menor de 10, se ejecuta la primera sentencia cont, en caso contrario se ejecuta la segunda sentencia cout.

Ejemplo

Combinación de dos operadores "or" (acentos ||)

$(y < 0) \ || \ (y > 12)$

Es verdad si y es menor que 0 o y es mayor que 12.

Ejemplo

Se puede negar cualquier expresión booleana utilizando el operador !. Si desea negar una expresión booleana, sitúe la expresión entre paréntesis y ponga el operador ! delante de ella.

$!(x < y)$  significa "x es no menor que y"

$!(x < y)$  es equivalente a  $x \geq y$

**Operador or**

Se puede formar una expresión booleana «or» combinando dos expresiones booleanas y utilizando el operador or (||).

$(\text{expresión\_booleana\_1}) \ || \ (\text{expresión\_booleana\_2})$

**Aplicación**

```
if ((x == 1) || (x == y))
    cont << "x es 1 o x es igual a y.\n";
else
    cont << "x ni es 1 ni igual a y.\n";
```

ahora te mostrare algunos ejemplos que nos permitan entender más sobre los operadores lógicos

**Ejemplos**

1. Realizar un algoritmo que permita verificar si un número está en un rango específico

Escribir "Ingrese un número:"

Leer numero

Si (numero  $\geq$  10 AND numero  $\leq$  20) entonces

Escribir "El número está en el rango de 10 a 20"

Sino

Escribir "El número está fuera del rango"

Fin Si

2. Realizar un algoritmo que me permita determinar si un año es bisiesto o si es un año nuevo

Escribir "Ingrese un año:"

Leer año

Si (año  $\% 4 = 0$  AND año  $\% 100 \neq 0$ ) OR (año  $\% 400 = 0$ ) entonces

Escribir "El año es bisiesto"

Sino si año = 2023 entonces

Escribir "¡Feliz año nuevo!"

Sino

Escribir "El año no es bisiesto y no es un año nuevo"

Fin Si

3. Realizar un algoritmo que me permita validar si un estudiante aprobó todas las asignaturas del colegio teniendo en cuenta que la nota aprobatoria es de 13 a más

Escribir "Ingrese la calificación de Matemáticas:"

Leer matematicas

Escribir "Ingrese la calificación de Ciencias:"

Leer ciencias

Escribir "Ingrese la calificación de Historia:"

Leer historia

Si (matematicas  $\geq$  60 AND ciencias  $\geq$  60 AND historia  $\geq$  60) entonces

Escribir "El estudiante aprobó todas las asignaturas"

Sino

Escribir "El estudiante no aprobó todas las asignaturas"

Fin Si



INSTITUTO  
**KHIPU**