

## Operadores relacionales

Son aquellos que permiten comparar expresiones. Si la evaluación es correcta, la máquina retorna VERDADERO, de lo contrario, retorna FALSO.

Ver : imagen01

Los operadores relacionales tienen menor prioridad que los aritméticos, esto significa que, por ejemplo, la siguiente expresión:  $5 + 4 < 3 - 2$ , será evaluada primero en términos de operadores aritméticos, quedando la siguiente expresión:  $9 < 1$ , donde se terminan de evaluar los operadores relacionales que hubiere (en este caso, solo uno), dando como resultado FALSO (9 no es menor que 1). Para comprobar como PSeInt evalúa las expresiones booleanas, te invito a que pruebes el siguiente programa, que cuenta con unas cuantas de ellas, de menor a mayor complejidad. Tu tarea no tan solo es ver los resultados de la ejecución del programa, sino que evalúes las expresiones vos también de manera manual para comprobar que el resultado es el mismo.

Ver :Imagen02

## Operadores lógicos

Son aquellos que permiten operar con expresiones booleanas. A través de ellos, pueden realizarse condiciones más complejas que no requieran un anidamiento de estructuras de selección. Así como los operadores aritméticos tienen como operandos a números, los operadores lógicos tienen como operandos a valores lógicos (VERDADERO o FALSO). Así como los operadores aritméticos devuelven como resultado un número, los operadores lógicos devuelven como resultado un valor lógico (VERDADERO o FALSO).

Ver imagen 03

### **-El operador NOT,**

también llamado NO o negación lógica, a diferencia de los que venías observando, es un operador monádico. Esto significa que trabaja con un solo operando. Lo que hace este operador es negar la expresión a la cual afecta. Se representa con el símbolo !.

Ver imagen04

## **-Operador OR**

El operador OR, también llamado O o disyunción lógica, trabaja con dos operandos. Lo que hace este operador es devolver un valor VERDADERO si al menos un operando es VERDADERO, en caso contrario devuelve FALSO. Se representa con el símbolo |. Es importante aclarar que, si bien PSeInt permite usar un solo |, en muchos lenguajes un operador | representa un OR de bajo nivel, es decir bit a bit, cuestión que no trataré, representando al operador OR lógico con un |. En este libro usaré el operador |. Voy con un ejemplo. La condición para pasar a cierto evento es ser mayor de edad o tener un nombre cuya longitud no supere los seis caracteres. Haré un programa que permita al usuario ingresar su nombre y su edad. Luego la máquina informará si puede pasar o no, según la condición planteada. El código puede ser el siguiente:

Ver imagen05

Una buena aplicación práctica de este operador podría ser para pedirle una respuesta al usuario. Suponé una aplicación que requiera que el usuario ingrese la palabra "Si" para continuar. Es evidente que puede ingresarlo todo en minúsculas, todo en mayúsculas o de forma alternada. Para contemplar todas estas combinaciones y que la máquina las interprete como la misma opción, podés armar una condición con operadores lógicos.

Ver imagen06

## **Operador AND**

El operador AND, también llamado Y o conjunción lógica, trabaja con dos operandos. Lo que hace este operador es devolver un valor VERDADERO si todos los operandos resultan VERDADERO, en caso contrario devuelve FALSO. Se representa con el símbolo &. Es importante aclarar que, si bien PSeInt permite usar un solo &, en muchos lenguajes un operador & representa un AND de bajo nivel, es decir bit a bit, cuestión que no trataré, representando al operador AND lógico con un &. En este libro usaré el operador &. Seguiré con el mismo ejemplo anterior, pero esta vez cambiaré las reglas. Fijate como cambiar un o por un y hace una gran diferencia. La condición para pasar a cierto evento es ser mayor de edad y tener un nombre cuya longitud no supere los seis caracteres. Haré un programa que permita al usuario ingresar su nombre y su edad. Luego la máquina informará si puede pasar o no, según la condición planteada.

Ver imagen07