

Tarea 14. Comparaciones LOGICAS.

Continuamos con comparaciones lógicas. Donde podremos comparar valores del sistema binario.

Utilizando operadores "and" y "or".

Para cada operador tenemos una tabla de verdad.

NOT	Negación !
0	1
1	0

AND		&& (=>dev c)
Posibilidades		Si todas sus entradas con 1
ENTRADAS		SALIDA
A	B	C
0	0	0
0	1	0
1	0	0
1	1	1

OR		(=>dev c)
Posibilidades		Si al menos una entrada es 1
ENTRADAS		SALIDA
A	B	C
0	0	0
0	1	1
1	0	1
1	1	1

Forma de utilizar las tablas.

Es importante reflexionar que se realiza la comparación par a par SOLAMENTE. Sin embargo, podemos imaginar los resultados de varios pares.

Es decir que pasaría si utilizo varias entradas y con "and". SIEMPRE tendré dos posibles soluciones:

- a) Si todas las entradas son cero, o si al menos EXISTE un cero en alguna entrada, su salida se invalida o me da falsa o meda CERO.
- b) Si todas las entradas son uno, mi salida será verdadera o 1.

Por otra parte, ¿que pasaría con varias entradas "or"? TAMBIEN tendré DOS opciones de resultado.

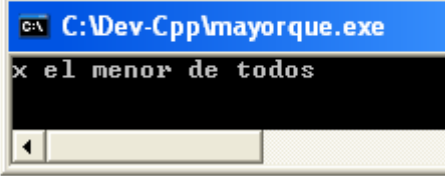
- a) Si todas las entradas son cero, su salida se invalida o me da falsa o CERO.
- b) Si aparece un solo 1 en un mundo de entradas 0, mi salida será verdadera o 1.

Negación es lo contrario (de un solo bit): si es 1 entonces CERO, si es CERO entonces el resultado es 1.

Empecemos a utilizar estas proposiciones.

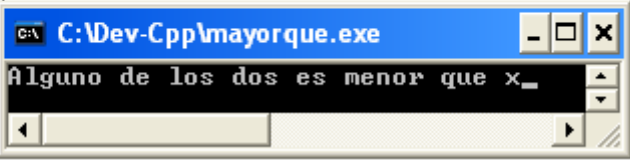
Primero utilizando "and" que en DEVCC es : &&

```
#include <stdio.h>
#include <conio.h>
int x,z,k;
main()
{
    x=2;
    z=4;
    k=7;
    if ( (x<z) && (x<k) )
        printf("x el menor de todos");
    else
        printf("Aun no puedo saber mas");
    getch();
}
```



Ahora utilizando "OR".

```
#include <stdio.h>
#include <conio.h>
int x,z,k;
main()
{
    x=2;
    z=1;
    k=7;
    if ( (x<z) || (x<k) )
        printf("Alguno de los dos es menor que x");
    else
        printf("x es mayor");
    getch();
}
```



Ahora utilicemos el valor NEGADO.

```

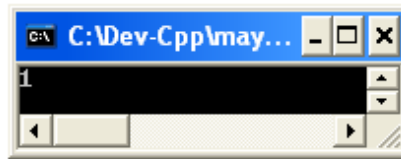
#include <stdio.h>
#include <conio.h>
int x, z, k;
main()
{
    x=0;
    z=1;

    if (! (x))
        {
            k=! (x) ;
            printf ("%d", k) ;
        }

    else
        {
            k= (x) ;
            printf ("%d", k) ;
        }

    getch() ;
}

```



Comparemos tres números. ¿Cuál es el menor?

```

#include <stdio.h>
#include <conio.h>
int x, z, k;
main()
{
    x=5;
    z=1;
    k=7;

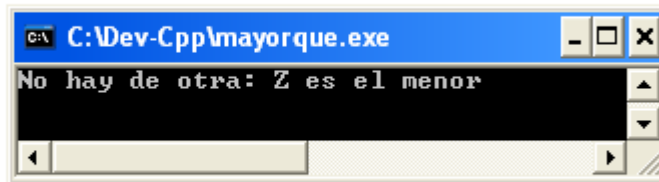
    if ((x<z) && (x<k))
        printf("No hay de otra: X es el menor");

    if ((z<x) && (z<k))
        printf("No hay de otra: Z es el menor");

    if ((k<z) && (k<x))
        printf("No hay de otra: K es el menor");

    getch() ;
}

```



Chequemos que nuestro método funciona para cualquier valor de cualquiera de las tres variables. Cambiemos valores. Y NO modifiquemos los IFs.

```

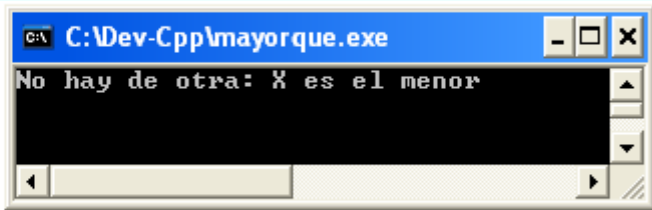
#include <stdio.h>
#include <conio.h>
int x, z, k;
main()
{
    z=5;
    x=1;
    k=7;

    if ((x<z) && (x<k))
        printf("No hay de otra: X es el menor");

    if ((z<x) && (z<k))
        printf("No hay de otra: Z es el menor");

    if ((k<z) && (k<x))
        printf("No hay de otra: K es el menor");
    getch();
}

```



Ahora otra vez cambiemos valores de las variables.

```

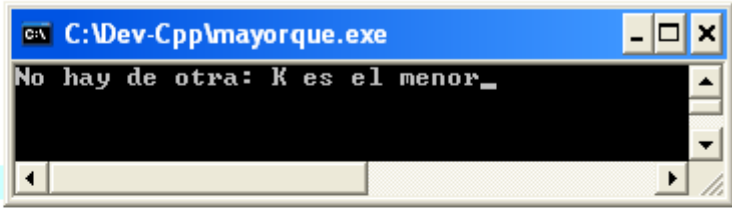
#include <stdio.h>
#include <conio.h>
int x, z, k;
main()
{
    z=5;
    k=1;
    x=7;

    if ((x<z) && (x<k))
        printf("No hay de otra: X es el menor");

    if ((z<x) && (z<k))
        printf("No hay de otra: Z es el menor");

    if ((k<z) && (k<x))
        printf("No hay de otra: K es el menor");
    getch();
}

```



Con estos tres ejemplos corroboramos que nuestro método de encontrar el menor de tres números esta bien.

Ejercicios:

13.1 Encuentra el mayor de tres números.

13.2 Encuentra cual es el numero que no es igual a ninguno de cuatro números.