

Solución Práctica 3: Condicionales.

1) Diga cuál será el resultado de los siguientes trozos de programa:

```
a. #include <stdio.h>
#define SI 1
#define NO 0
void main ( )
{
    int x, y, a;
    x = 5;
    y = 10;
    a = SI;
    if (a && x != y)
    {
        x = y - 1;
        y = x * y;
    }
    else
        x = x+y;
    printf ( "x = %d  y = %d", x,y);
}
```

SOLUCION. Escribirá: x=9 y=90

```
b. #include <stdio.h>
#define SI 1
#define NO 0
void main ( )
{
    int siono, x, y, z;

    siono = SI;
    x = 4;
    y = 8;
    switch (siono) {
        case SI : z = 2*x; break;
        case NO : z = y; break;
    }
    if (z == y)
        printf ("z = y\n");
    printf ("z = %d", z);
}
```

SOLUCION. Escribirá: z=y
z=8

2) Dado el siguiente programa en lenguaje C, diga qué hace en cada una de las opciones 1, 2, 3 y 4

```
#include <stdio.h>
main() {
    int Opc;
    int X;
    printf("Opciones: \n");
    printf("(1) \n");
    printf("(2) \n");
    printf("(3) \n");
    printf("(4) \n");
    printf("Introduzca la opcion:");

    scanf("%i",&Opc);

    printf("\nIntroduzca X:");
    scanf("%d",&X);
    switch(Opc) {
        case 1: X*=X;
            break;
        case 2: X=-X;
            break;
        case 3: X+=X;
            break;
        case 4: X%=2;
            break;
        default :printf("\nOpcion erronea\n");
    }
    if (Opc==1||Opc==2||Opc==3||Opc==4)
    printf("\nRespuesta = %d",X);
    return 0;
}
```

SOLUCIÓN

La opción 1 calcula el cuadrado de X

La opción 2 calcula el opuesto de X

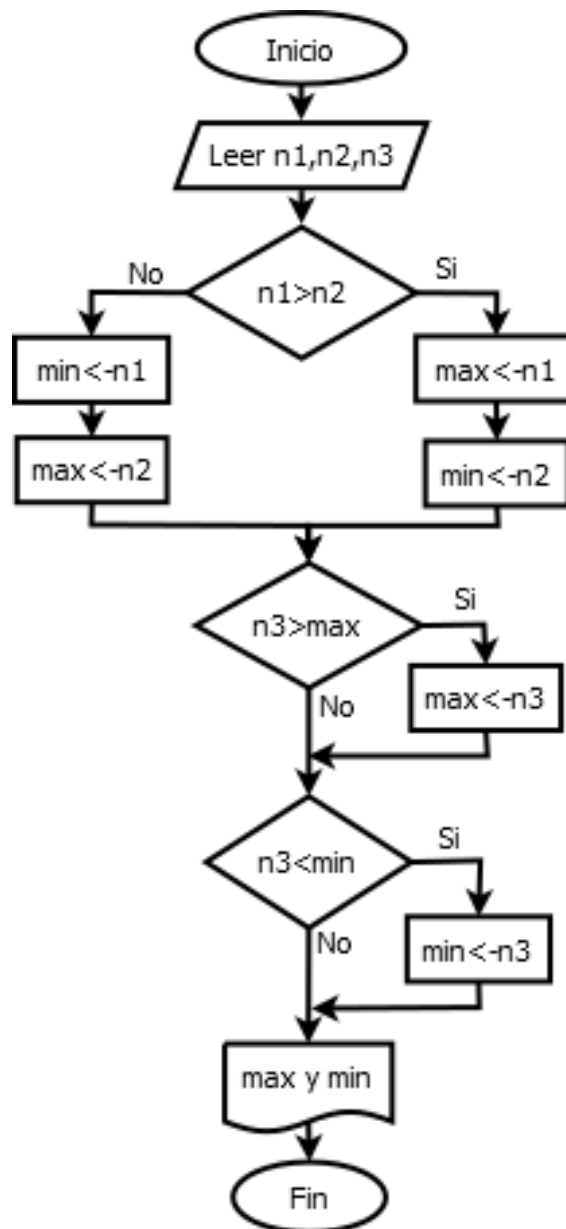
La opción 3 calcula el doble de X

La opción 4 calcula el residuo de dividir X entre 2

- 3) Escriba un algoritmo que dados tres números enteros, calcule el mayor y el menor de ellos.
Traduzca el algoritmo a lenguaje C.

ALGORITMO

ENTRADAS: n_1, n_2, n_3 (enteros)
// PRE: ninguna
SALIDAS: max, min (enteros)
// POST: $max \geq n_1 \wedge max \geq n_2 \wedge max \geq n_3 \wedge$
// $min \leq n_1 \wedge min \leq n_2 \wedge min \leq n_3$



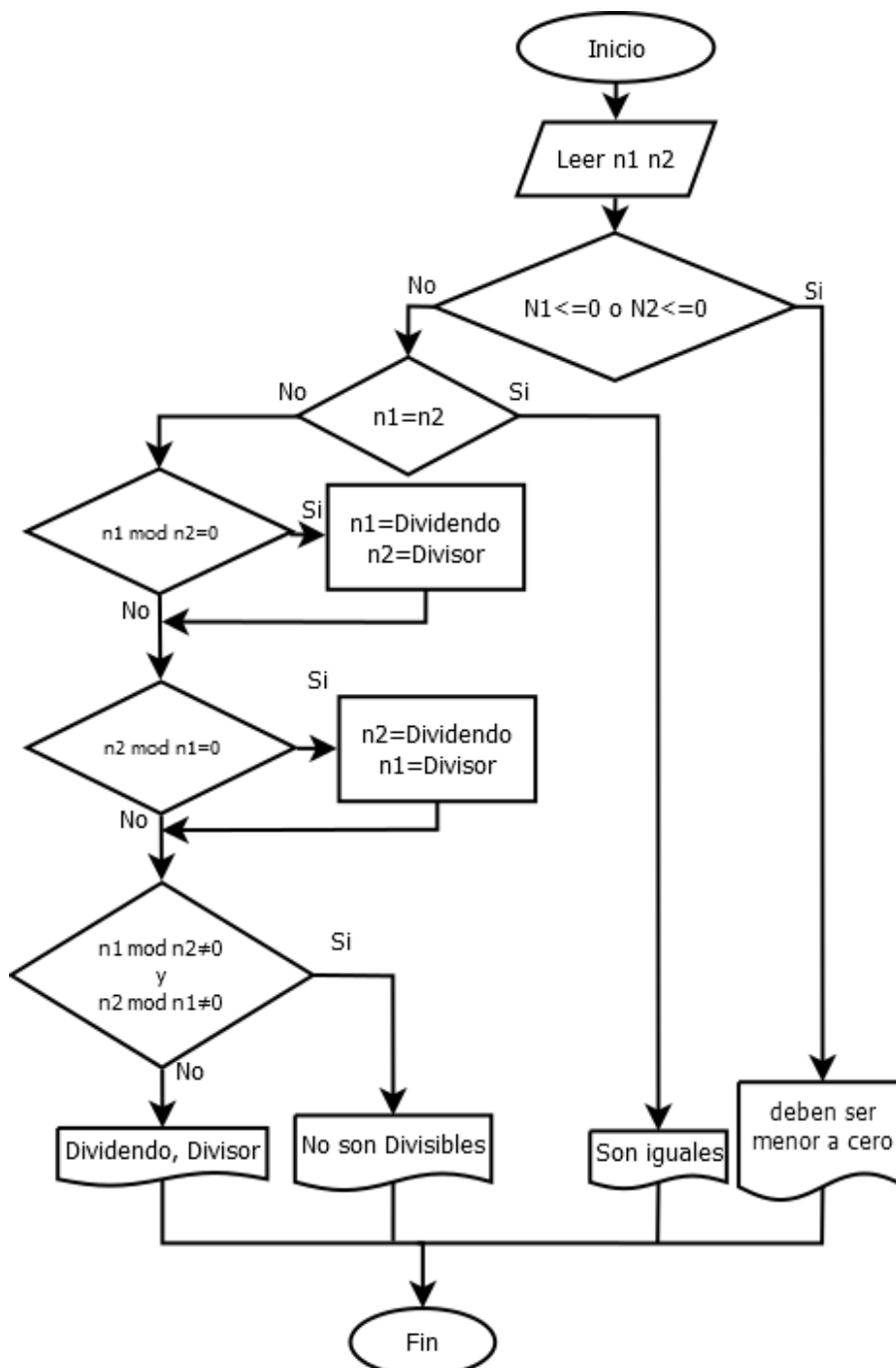
- 4) Escriba un algoritmo que dados dos números enteros no negativos, determine si alguno de ellos divide al otro. En caso afirmativo, debe imprimir cuál de ellos es el dividendo y cuál es el divisor, en caso negativo imprimir un mensaje de respuesta. Escriba el programa equivalente en C. Verifique las precondiciones usando condicionales.

ALGORITMO

```

ENTRADAS: n1,n2 (enteros)
// PRE: n1≥0  ^ n2≥0
SALIDAS: dividendo,divisor (enteros)
// POST: dividendo mod divisor = 0 ^
//       ((dividendo = n1 ^ divisor = n2) ∨
//       (dividendo = n2 ^ divisor = n1))

```



- 5) Escriba un algoritmo para determinar si un año es bisiesto. Recordar que un año es bisiesto si es divisible por 4, excepto aquellos divisibles por 100 pero no por 400. Escriba el programa equivalente en C. Verifique las precondiciones usando condicionales.

ALGORITMO

ENTRADAS: año (número natural)

// PRE: año ≥ 0

SALIDAS: bisiesto (lógico)

// POST: bisiesto \Leftrightarrow (año mod 400 = 0) \vee

// (año mod 4 = 0 \wedge año mod 100 $\neq 0$)

