

MÁXIMO MÚLTIPLO

BRIGADA 6

Análisis, describiendo el problema e identificando los datos de entrada y de salida “máximo múltiplo”.

Al realizar el programa de máximo múltiplo, nos indican que dado un divisor y un límite, encontrar el entero n más grande tal que.

Por ejemplo:

divisor = 2

límite = 3

num = 2

divisor = 5

límite = 40

num = 40

En cuanto a las librerías que utilizamos fue:

- **Stdio.h:** Esta biblioteca contiene las definiciones de macros, las constantes, las declaraciones de funciones y la definición de tipos usados por varias operaciones estándar de entrada y salida. Por otro lado, es el archivo de cabecera que contiene las definiciones de las macros, las constantes, las declaraciones de funciones de la biblioteca estándar del lenguaje de programación C para hacer operaciones, estándar, de entrada y salida, así como la definición de tipos necesarias para dichas operaciones.

Empezamos a declarar con **int main** como entrada de nuestro programa, declarado la función del tipo int, al llegar al final de su ejecución tendremos que devolver un valor. Normalmente se devuelve 0 para indicar que el programa ha finalizado correctamente, y cualquier otro valor para indicar una ejecución anormal.

La función main es imprescindible en cualquier programa C/C++ representa el punto de inicio de su ejecución. Así pues, main representa el punto de la ejecución a partir del cual el programador toma el control de la ejecución, antes de esto ya han sucedido muchas cosas en nuestro programa-

Después se declara con **int** las variables de numero como “num”, divisor como “divisor” y a limite como “limite”.

Mediante la función **printf** podemos escribir datos en el dispositivo de salida estándar. Además, transporta datos desde la memoria a la pantalla, a diferencia de scanf, que envía datos desde el teclado para almacenarlos en la memoria. La función printf devuelve el número de caracteres escritos. Si devuelve un valor negativo indica que se ha producido un error.

En la función **scanf** representa a una familia de funciones que analizan una entrada de datos con formato y cargan el resultado en los argumentos que se pasan por referencia a dicha función o funciones:

La función scanf() lee los datos de entrada en el stdin (flujo de entrada estándar).

La función fscanf() (file-scanf) lee en un flujo de entrada dado, por lo general un fichero (file) abierto para lectura.

Después tenemos a la sentencia **if-else** del cual es usada para ejecutar una instrucción o bloque de instrucciones solo si una condición es cumplida. Donde la condición es la expresión que será evaluada. Si esta condición es true (verdadera), el extracto es ejecutado.

Dentro de la condición if tenemos definidos a limite y divisor en donde el limite debe ser mayor y el divisor menor para poder hacer la operación, después a numero con igualdad con limite

A continuación con un **else** del cual indica de lo contrario si el limite es menor y el divisor mayor no podrá realizarse la operación.

Finalmente volvemos a utilizar printf para indicar el máximo múltiplo y por ultimo para regresar al valor de cero.

.

Pseudocódigo

INICIO

num:ENTERO

divisor:ENTERO

limite:ENTERO

ESCRIBIR "Introduce un divisor"

LEER divisor

ESCRIBIR "Introduce algun límite"

LEER limite

SI

$\text{limite mod divisor} = 0$ ENTONCES $\text{num} := \text{limite}$

FIN SI

DE LO CONTRARIO

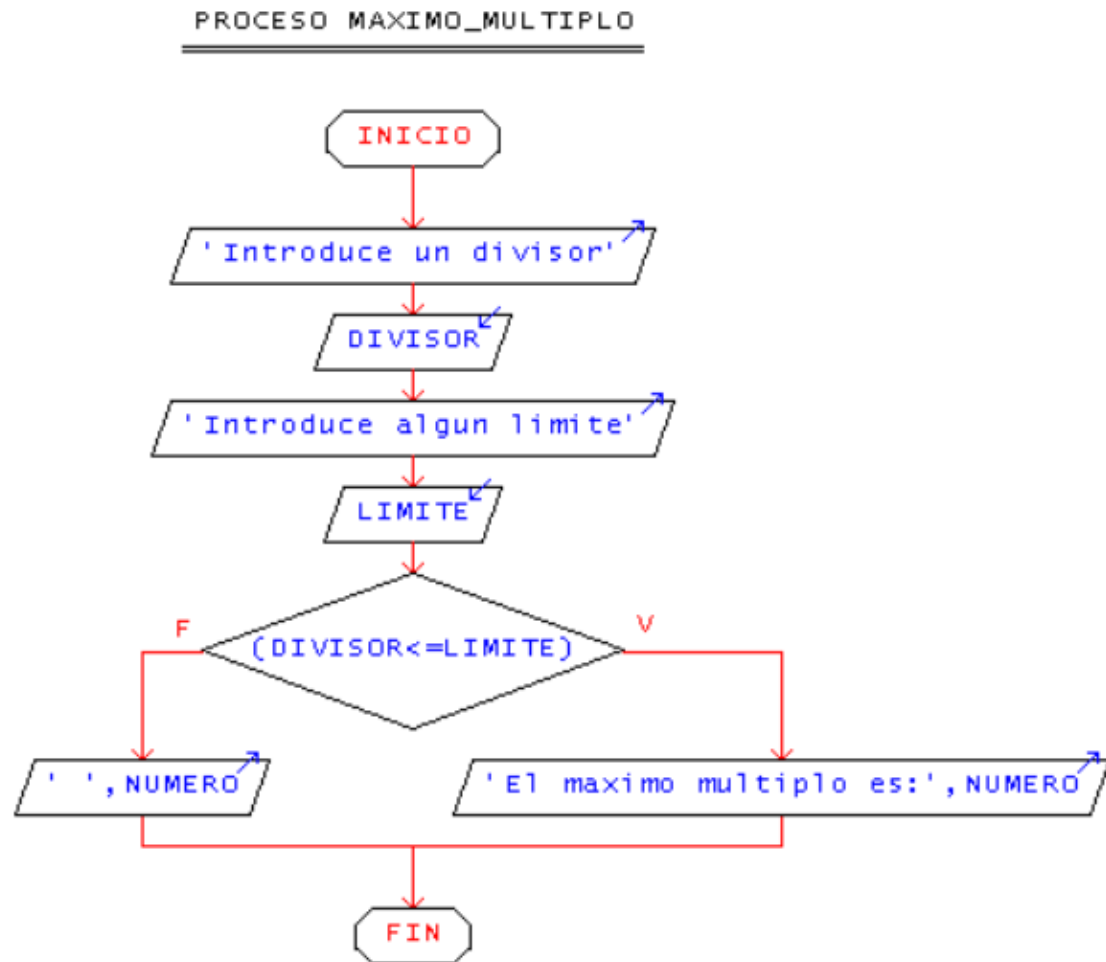
$\text{numero} := \text{limite} - (\text{limite mod divisor})$

FIN DE LO CONTRARIO

ESCRIBIR "El máximo múltiplo es:", num

FIN

Diagrama de flujo



Programa en C

```
#include <stdio.h>

int main()

{

    int num, divisor, limite;

    printf ("Introduce un divisor\n");

    scanf ("%d",& divisor);

    printf ("Introduce algún limite\n");

    scanf ("%d",& limite);

    if ((limite % divisor) == 0)

        num = limite;

    else

    {

        num=limite-(limite % divisor);

    }

    printf ("El maximo multiplo es: %d",num);

    return 0;

}
```

PRUEBA DE ESCRITORIO

```
Introduce un divisor
3
Introduce algún limite
20
El maximo multiplo es: 18
Process returned 0 (0x0)   execution time : 32.866 s
Press any key to continue.
```

TEST

/*

int divisor = 2;

int limite = 5;

generarMaximoMultiplo(divisor, limite), 4

int divisor = 5;

int limite = 30;

generarMaximoMultiplo(\$divisor, \$limite), 30

```

*/

#include <stdio.h>

int main()
{
    int n;
    int divs = 2;
    int lim = 5;
    if ((lim % divs) == 0)
        n = lim;
    else{
        n=lim-(lim % divs);
    }
    if (n==4)
    {
        printf("Ok\n");
        printf("Divisor: %d\n",divs);
        printf("Limite: %d\n",lim);
    }
    else
    {
        printf("Error\n");
        return 0;
    }
    int x;
    int divs1 = 5;

```



```
int lim1 = 30;

if ((lim1 % divs1) == 0)

    x = lim1;

else{

    x=lim1-(lim1 % divs1);

}

if (x==30)

{

    printf("Ok\n");

    printf("Divisor:\%d\n",divs1);

    printf("Limite:\%d",lim1);

}

else{

    printf("Error\n");

    return 0;

}

}
```