|  |  |
| --- | --- |
| Facultad de Ingeniería | Laboratorio de docencia |

Laboratorios de computación

salas A y B

Profesor: Karina García Morales

Asignatura: Fundamentos de Programación

Grupo: 20

No de práctica: 07

Integrantes: Leonardo Ponce Vivas

No de lista o brigada: 40

Semestre: 2021-1

Fecha de entrega: 25 de noviembre

Observaciones:

CALIFICACION:

Fundamentos de Lenguaje C

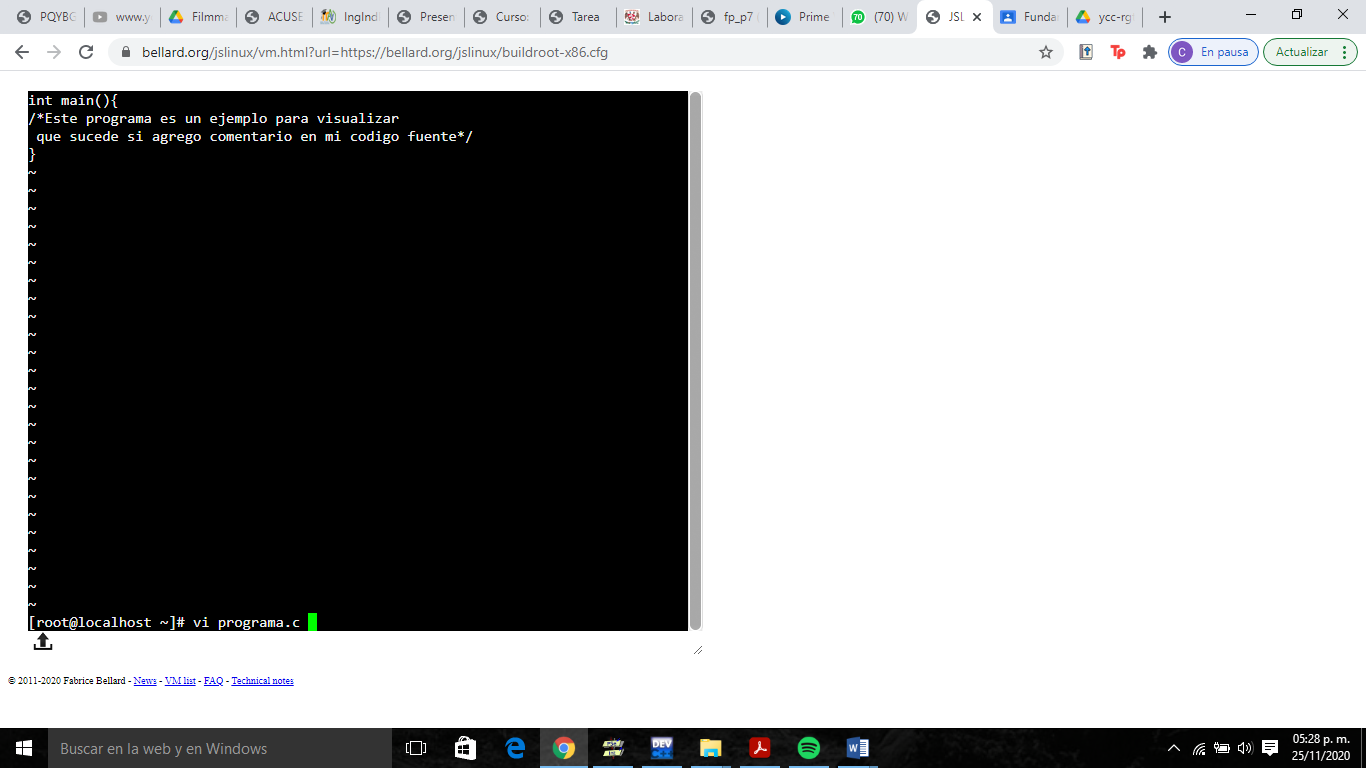
**Objetivo:**

Elaborar programas en lenguaje C utilizando las instrucciones de control de tipo secuencia, para realizar la declaración de variables de diferentes tipos de datos, así como efectuar llamadas a funciones externas de entrada y salida para asignar y mostrar valores de variables y expresiones.

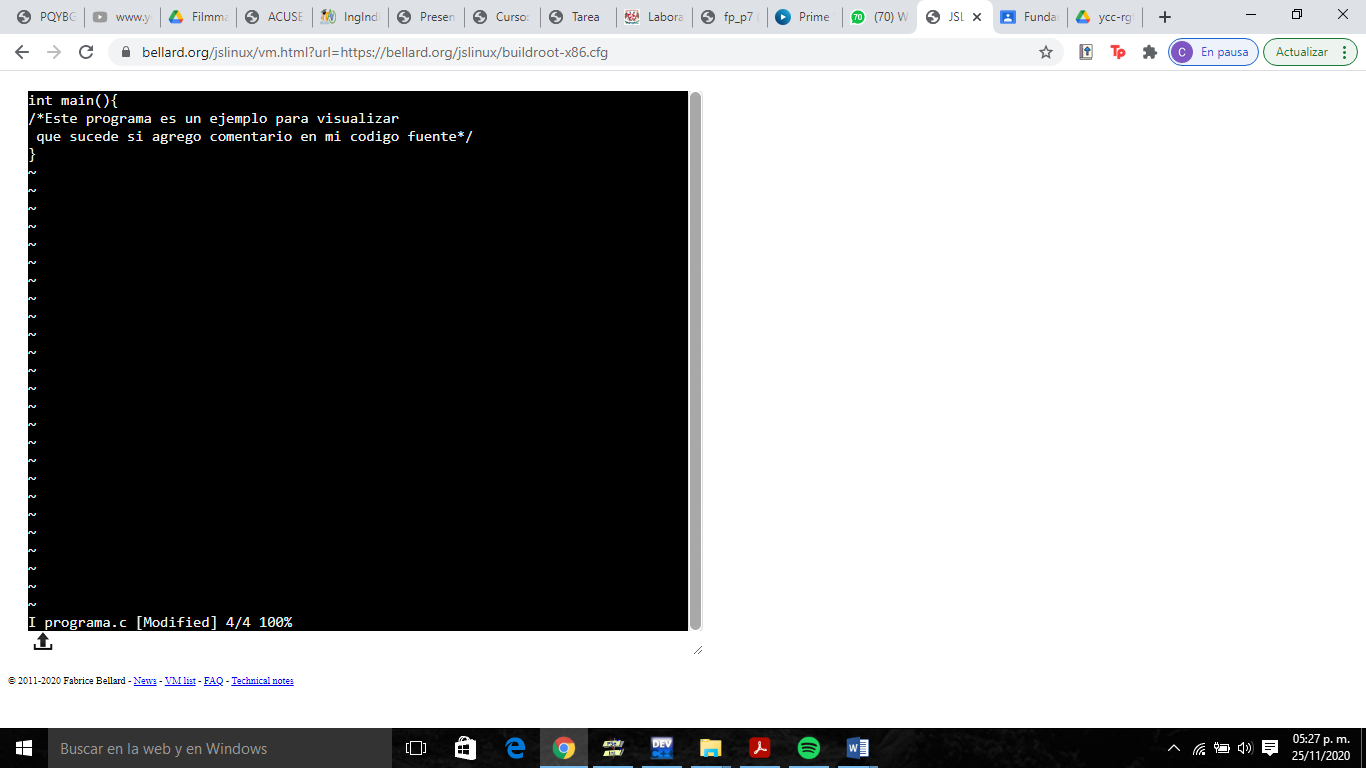
**Desarrollo:**

Una vez que un problema dado ha sido analizado, que se ha diseñado un algoritmo que lo resuelva de manera eficiente y que se ha representado el algoritmo de manera gráfica o escrita se procede a la etapa de codificación. La codificación se puede realizar en cualquier lenguaje de programación estructurada, como lo son Pascal, Python, Fortran o PHP. En esta ocasión se usará el lenguaje de programación C.

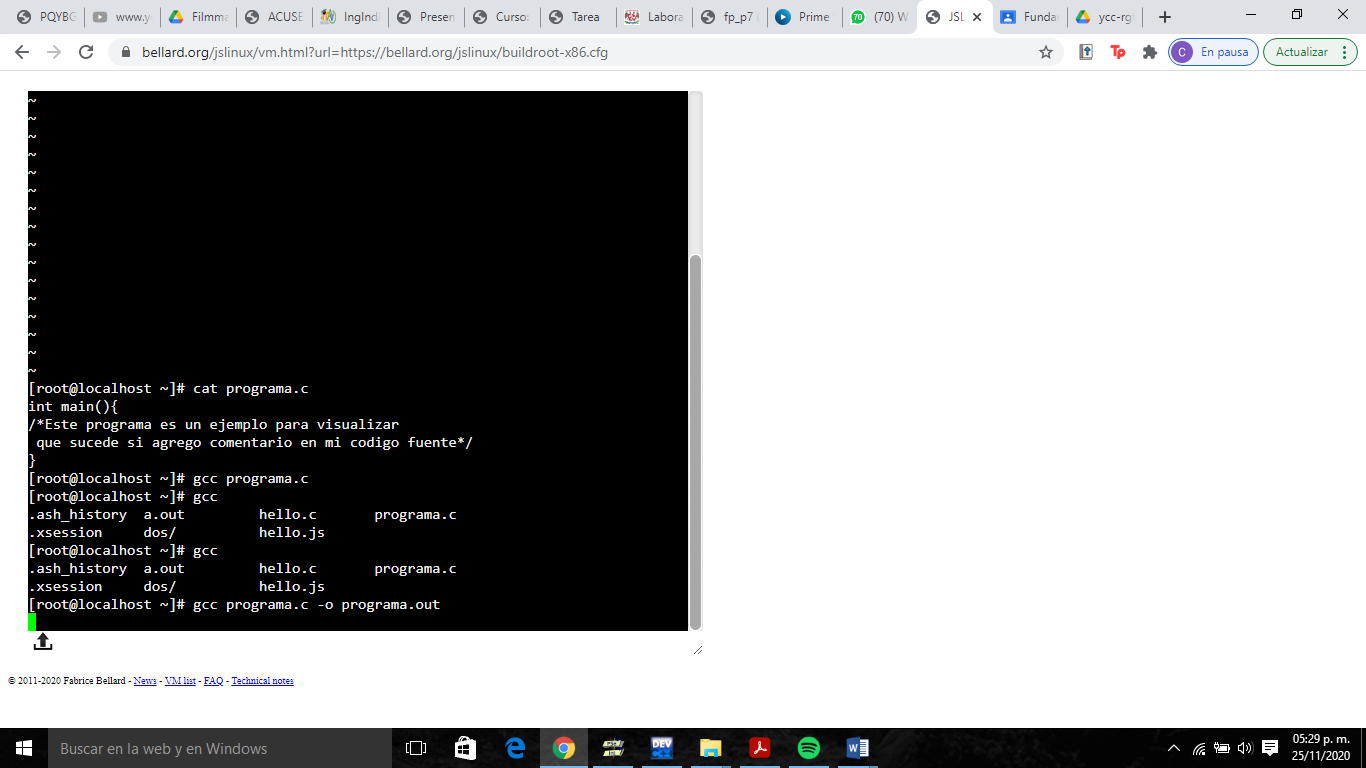
1.- Ejercicio



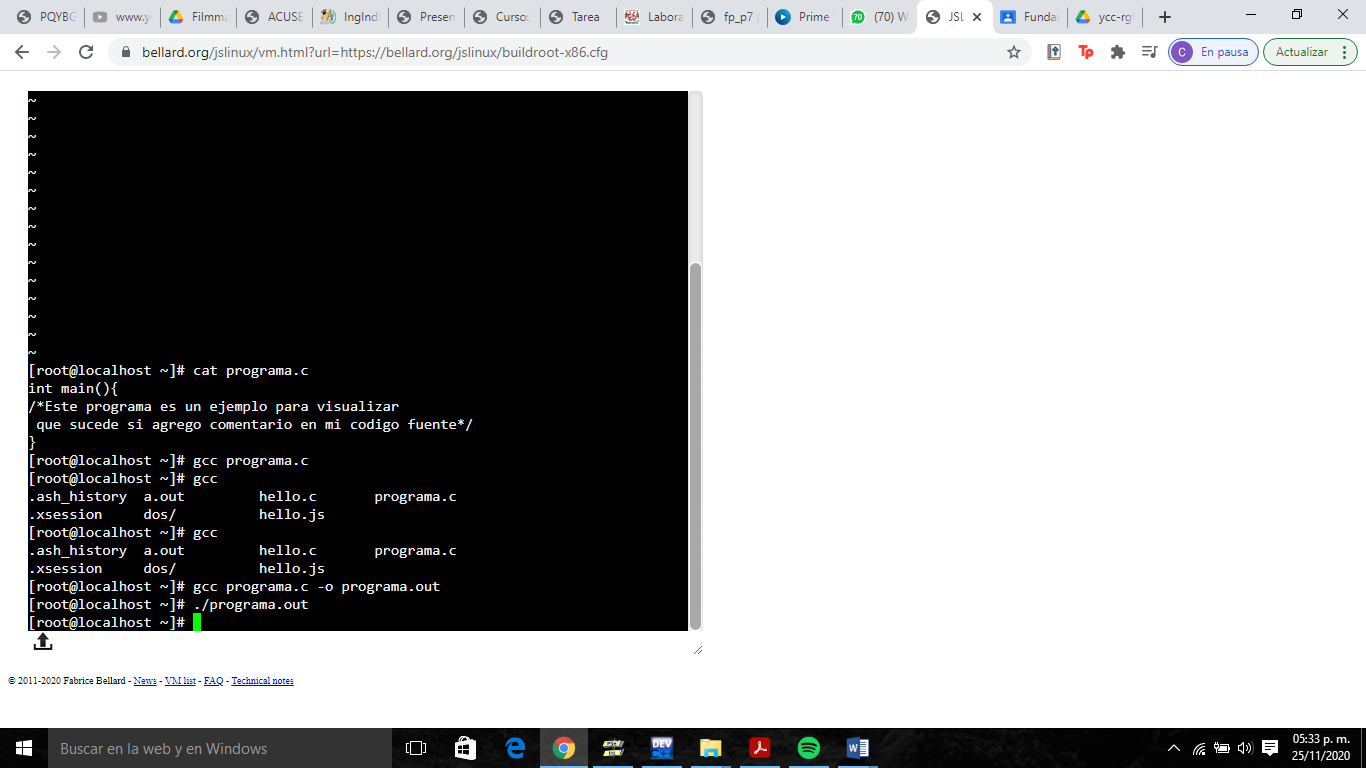
Creamos un archivo



Editamos programa.c

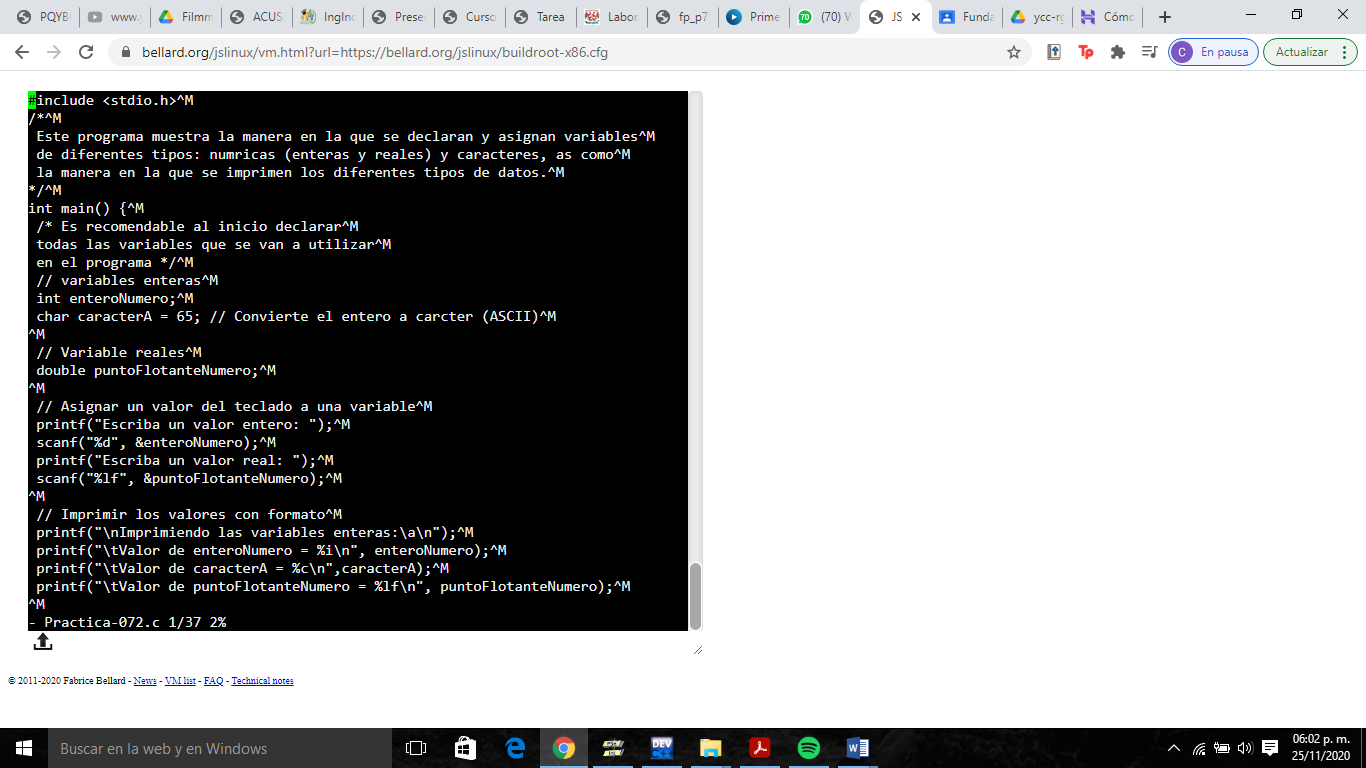


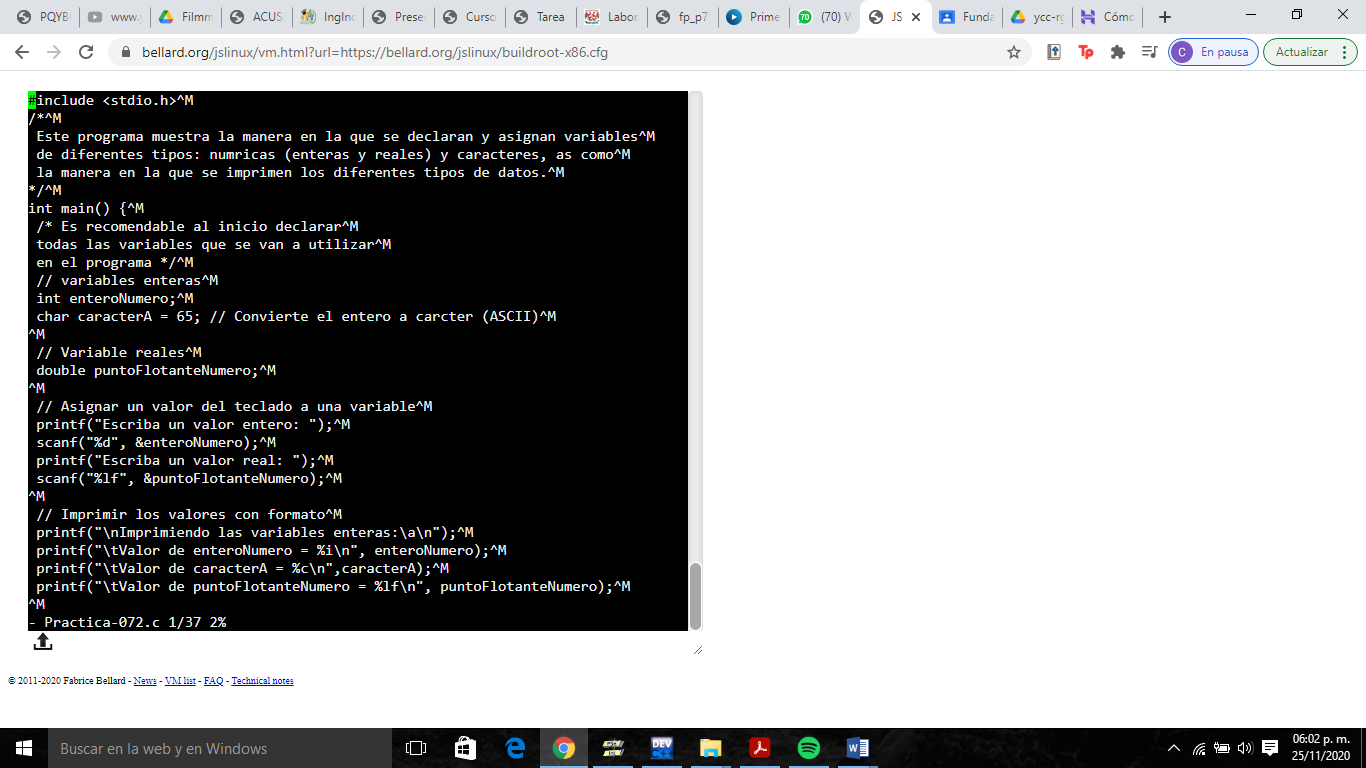
Compilamos programa.c



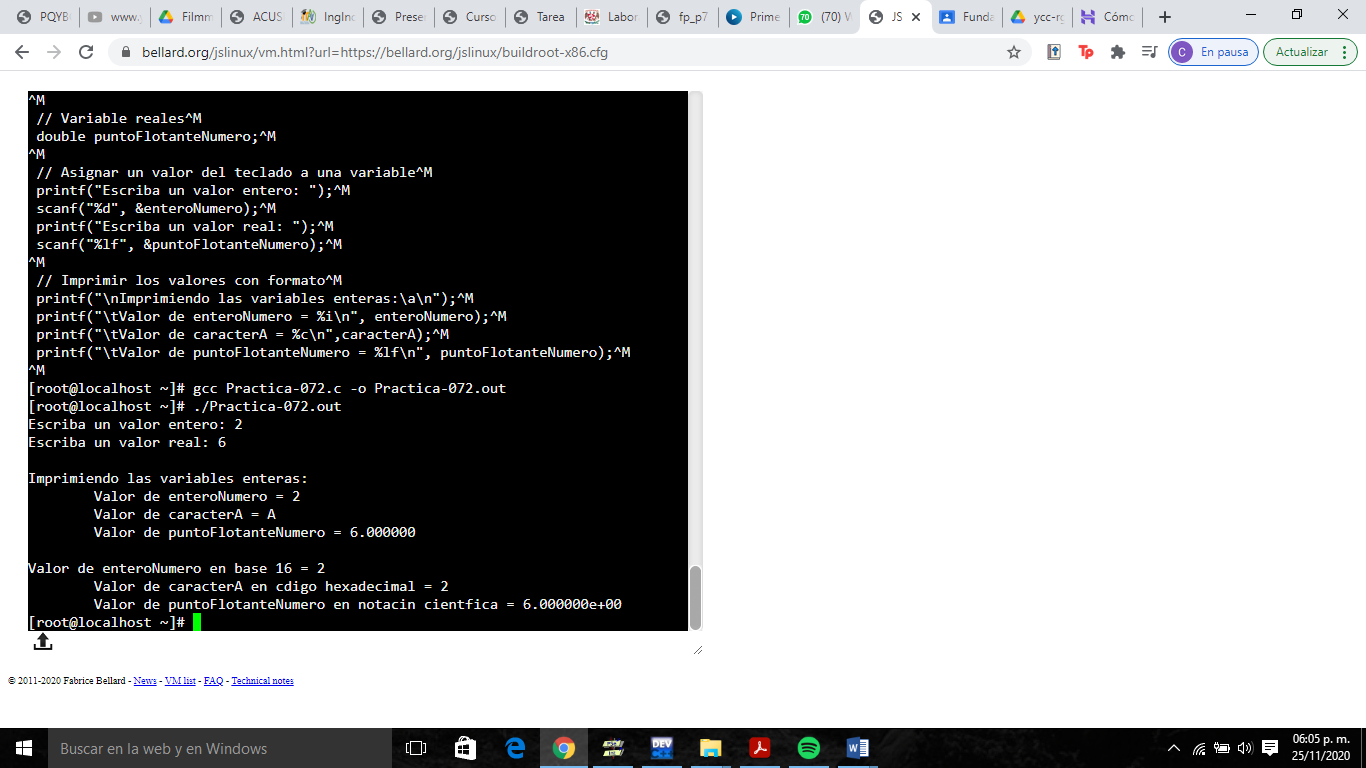
Y lo ejecutamos

1.1.- Ejercicio



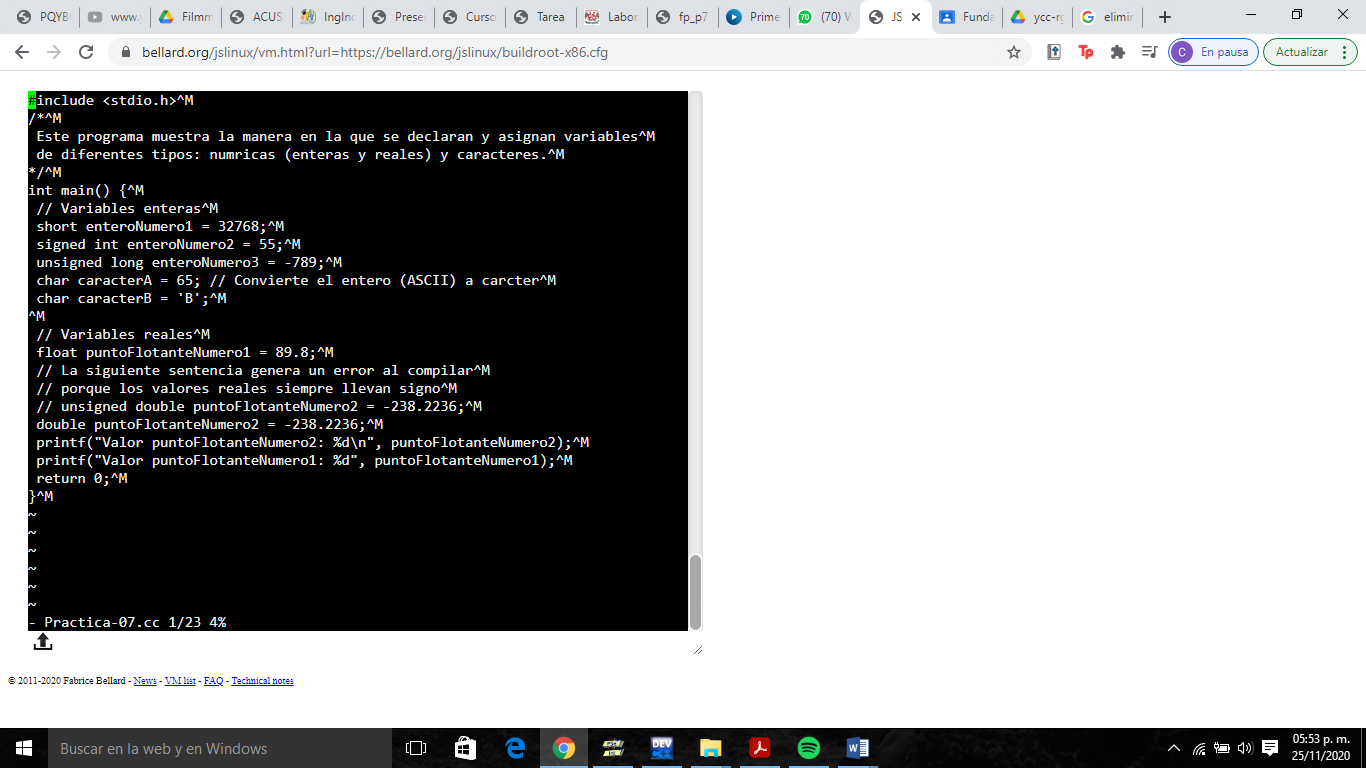


Cambiamos “CharA” por “ caracterA”



Lo compilamos y ejecutamos

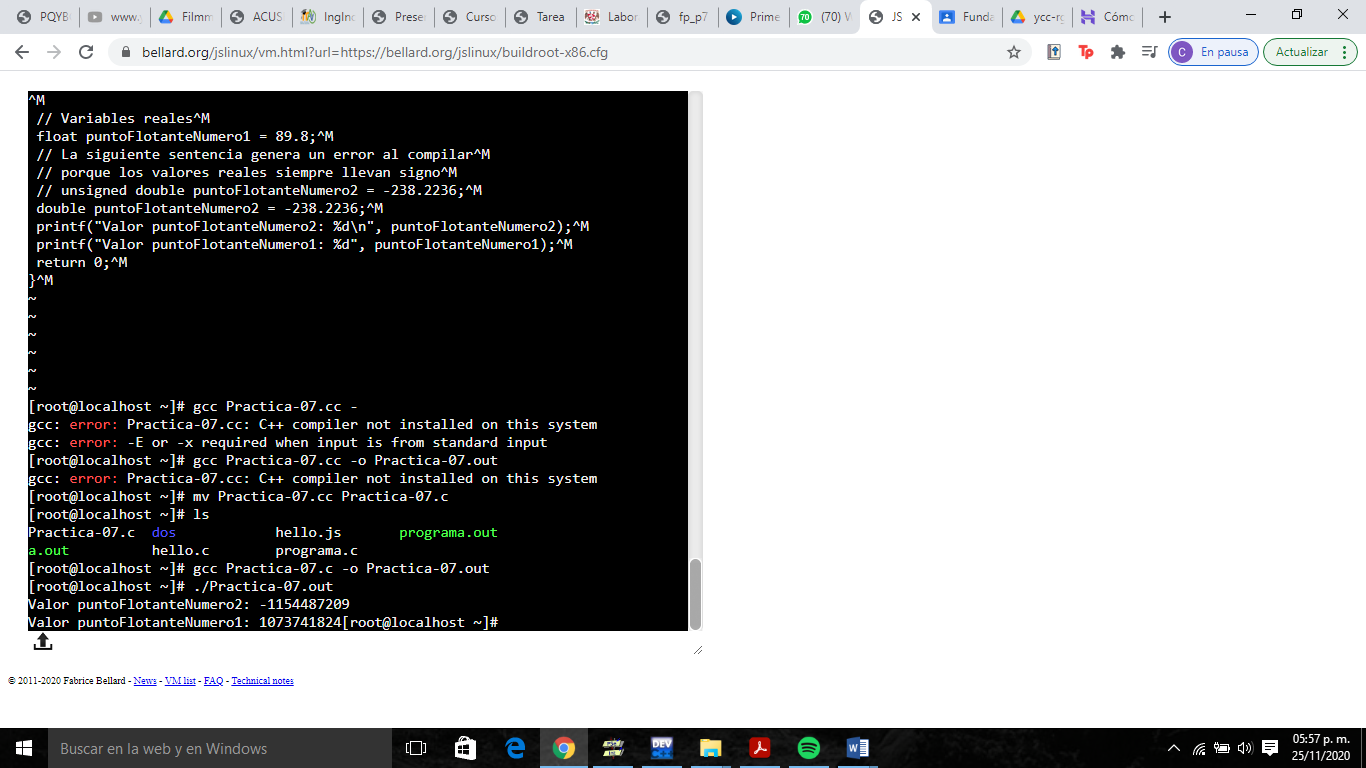
2.- Ejercicio



Se le agregaron los printf

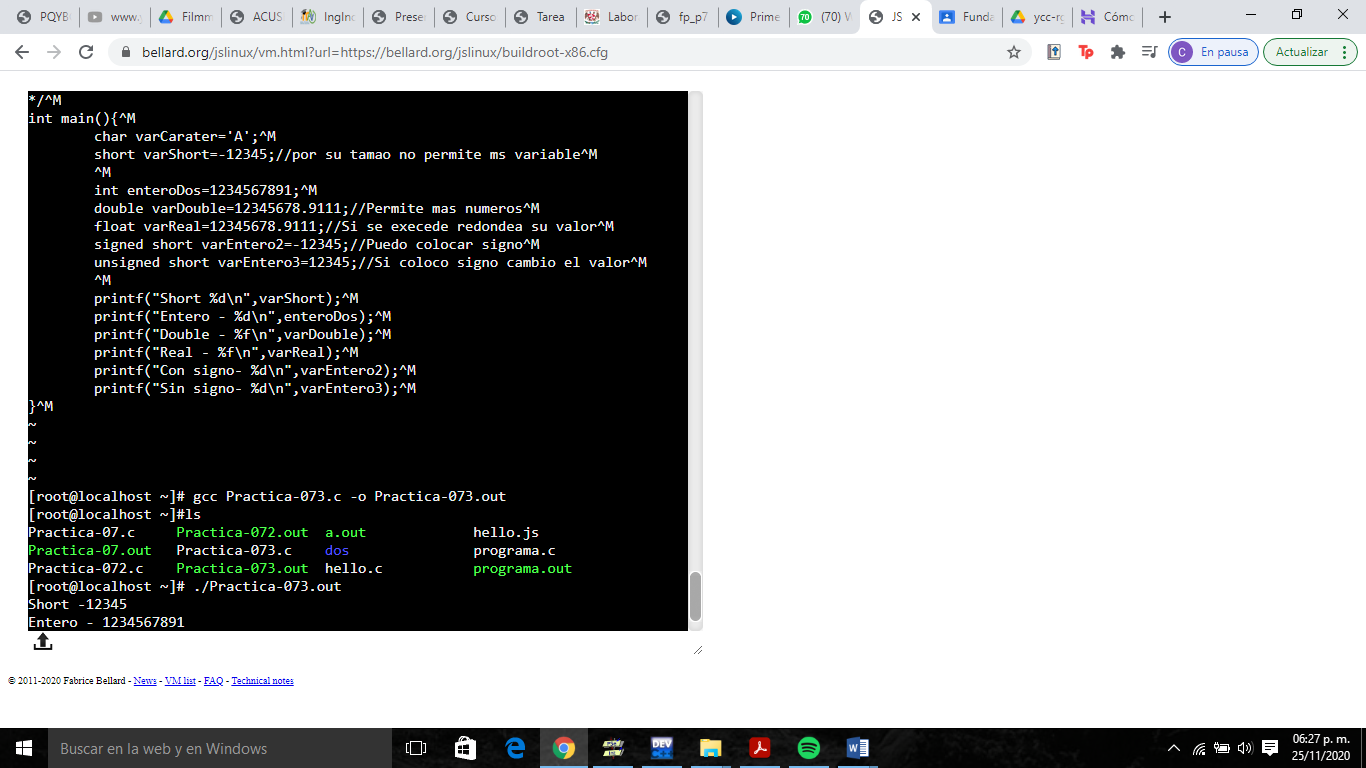
- printf("Valor puntoFlotanteNumero2: %d\n", puntoFlotanteNumero2);

-printf("Valor puntoFlotanteNumero1: %d", puntoFlotanteNumero1);

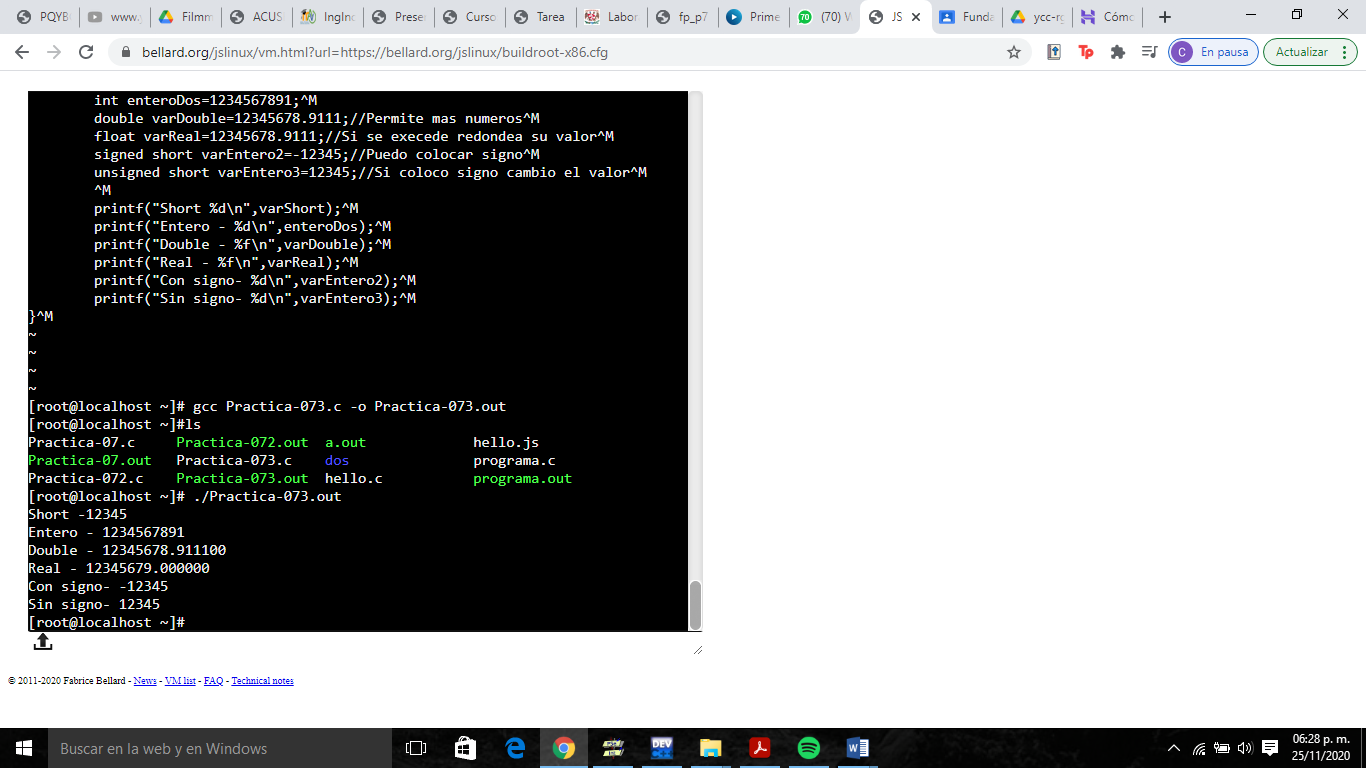


Lo compilamos y ejecutamos

3.- Ejercicio

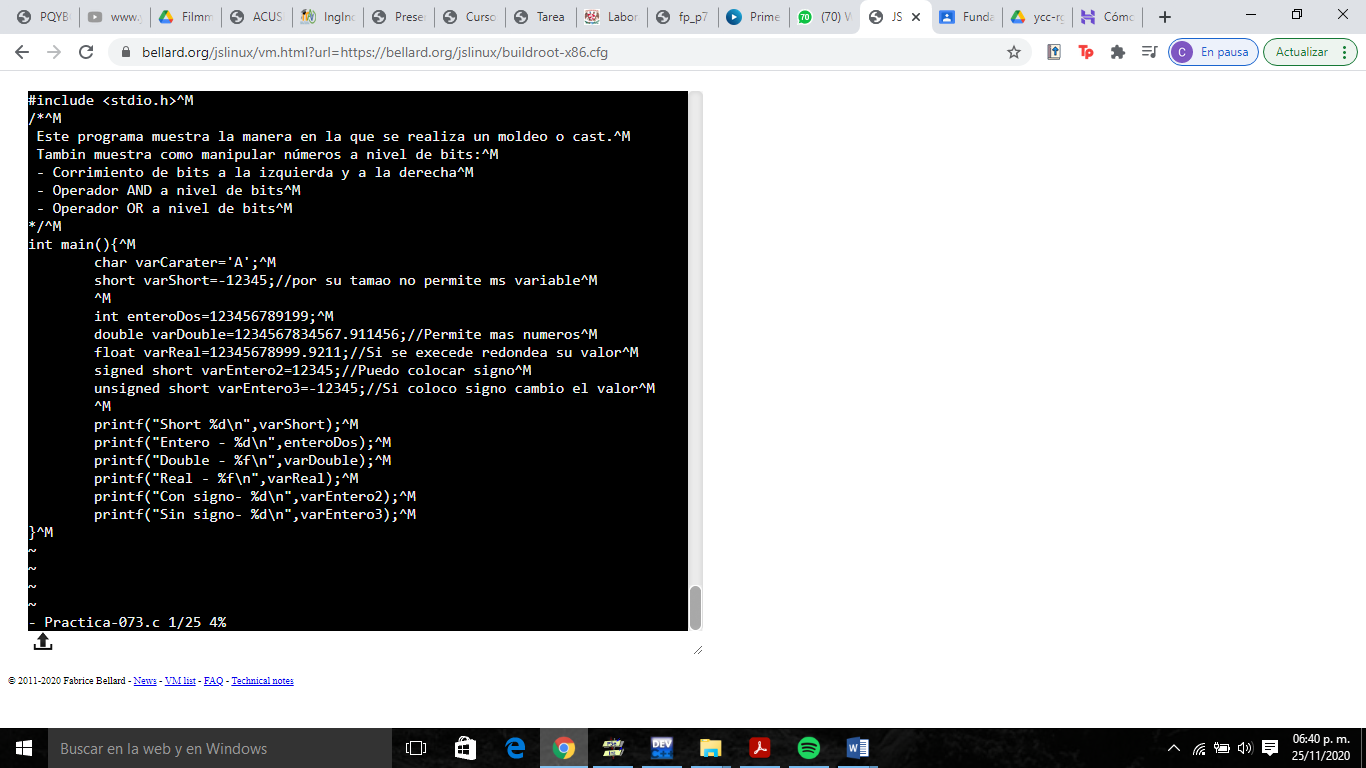


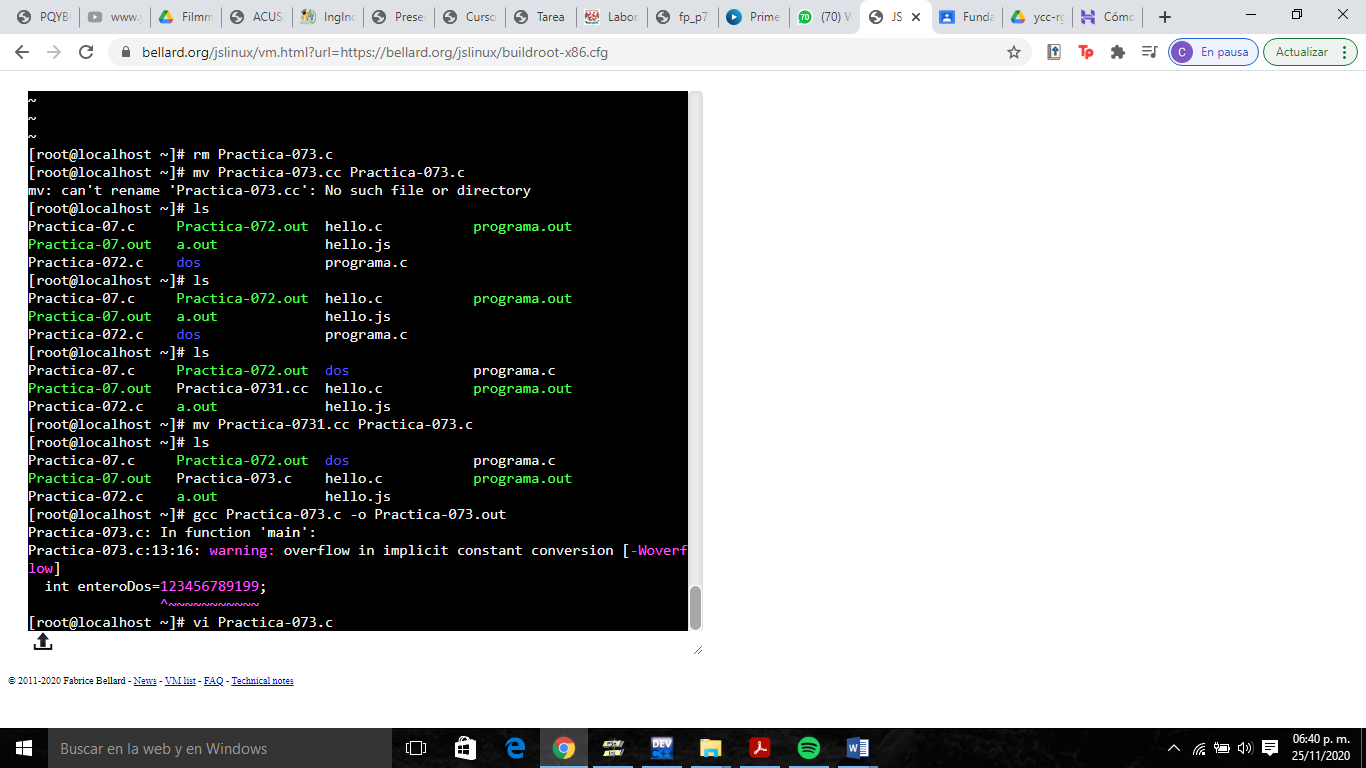
Agregamos un archivo

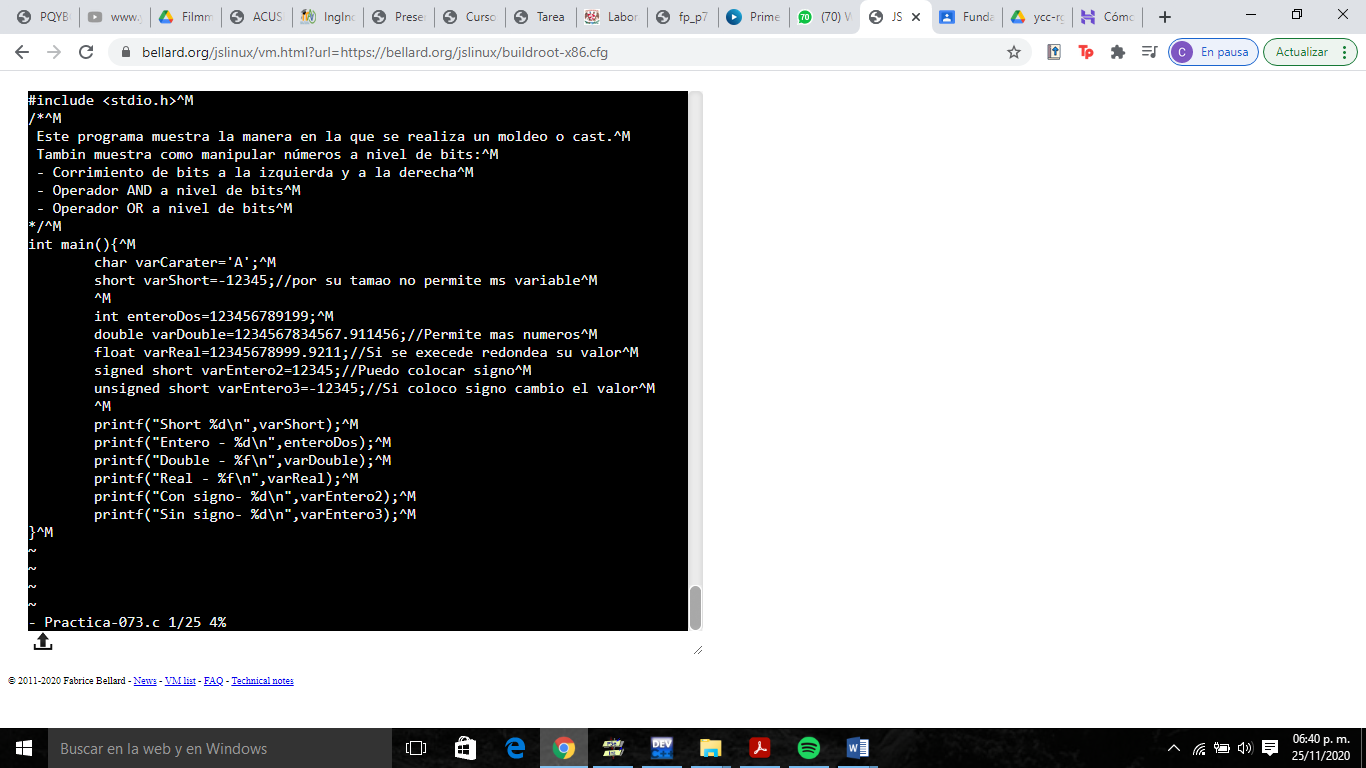


Lo compilamos y ejecutamos

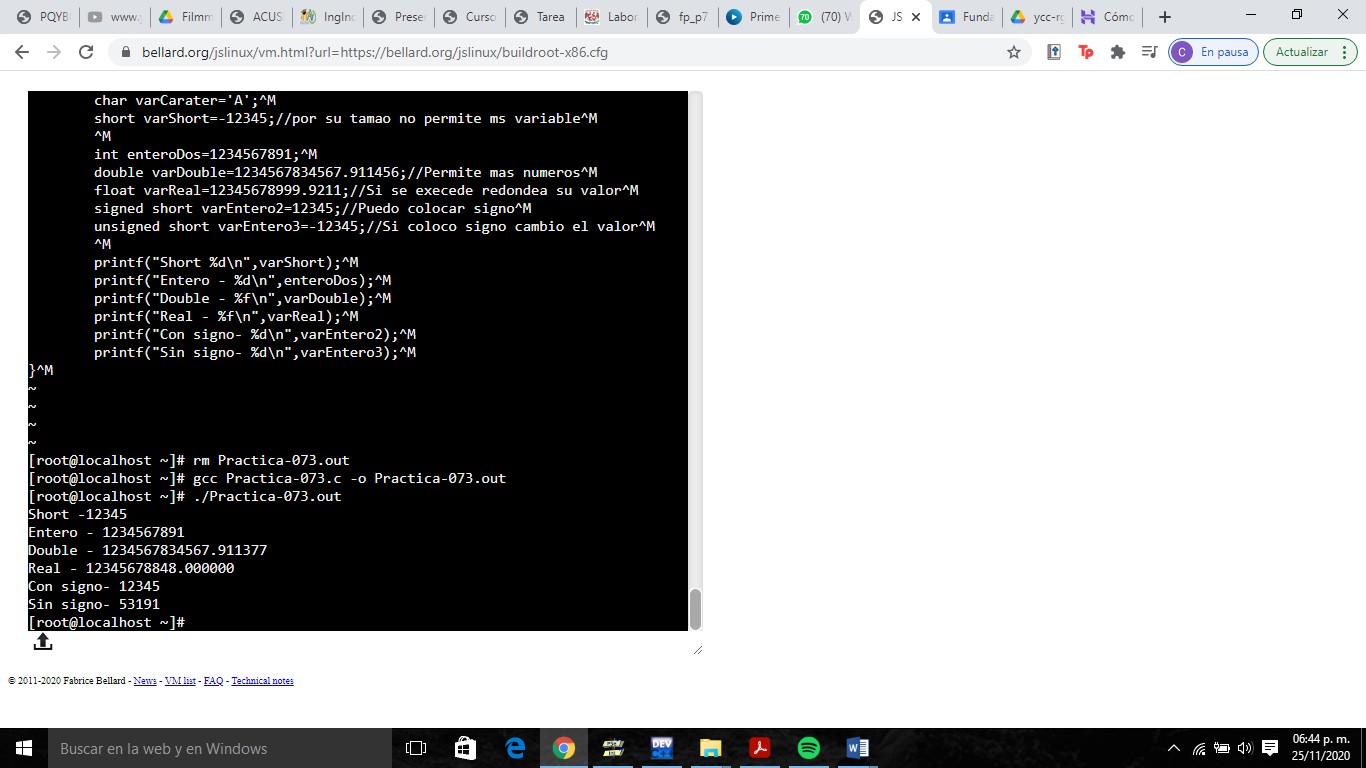
\*Modificar valores







Problema con enteroDos porque tenemos más valores de los permitidos



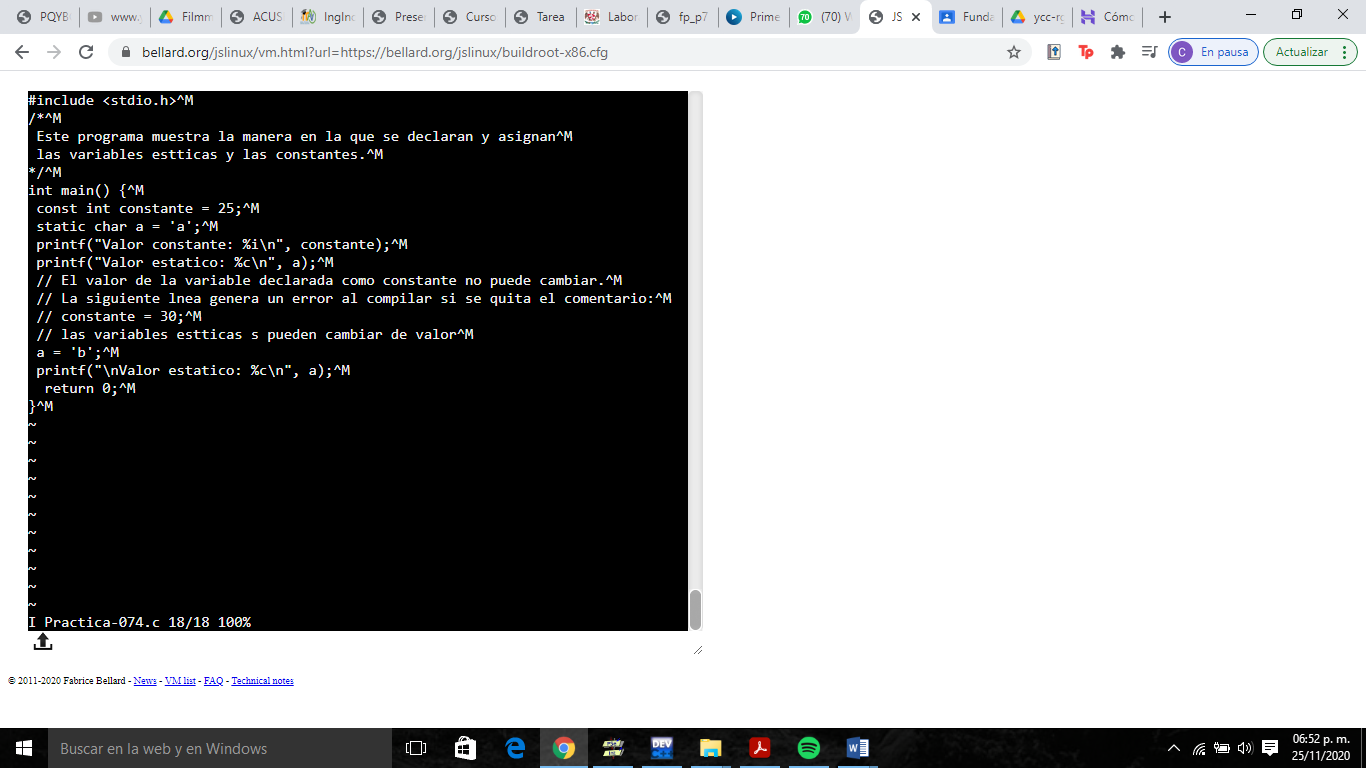
Modificamos y ejecutamos

Tenemos Con signo positivo porque se cambió de signo

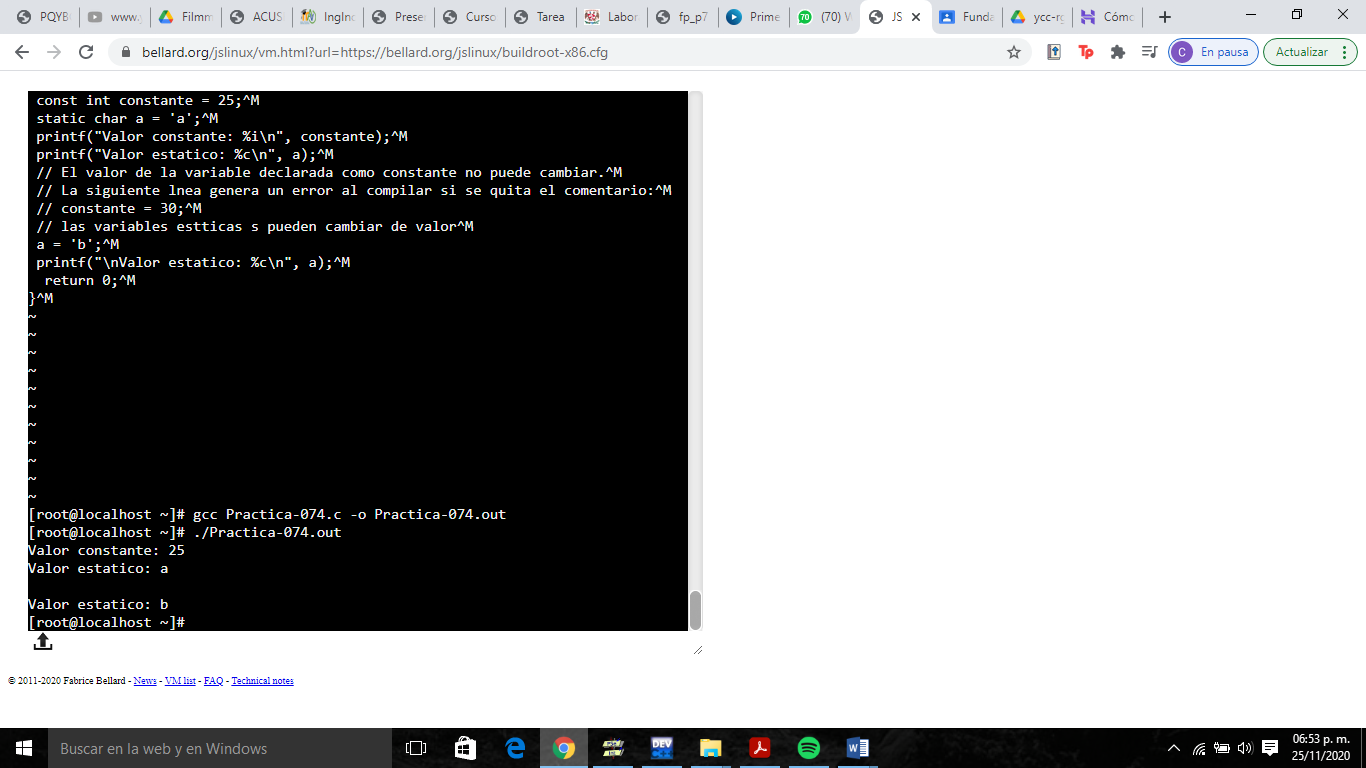
Tenemos Sin signo diferente porque se cambió de signo

Tenemos Double diferente porque se cambió de valor

4.- Ejercicio



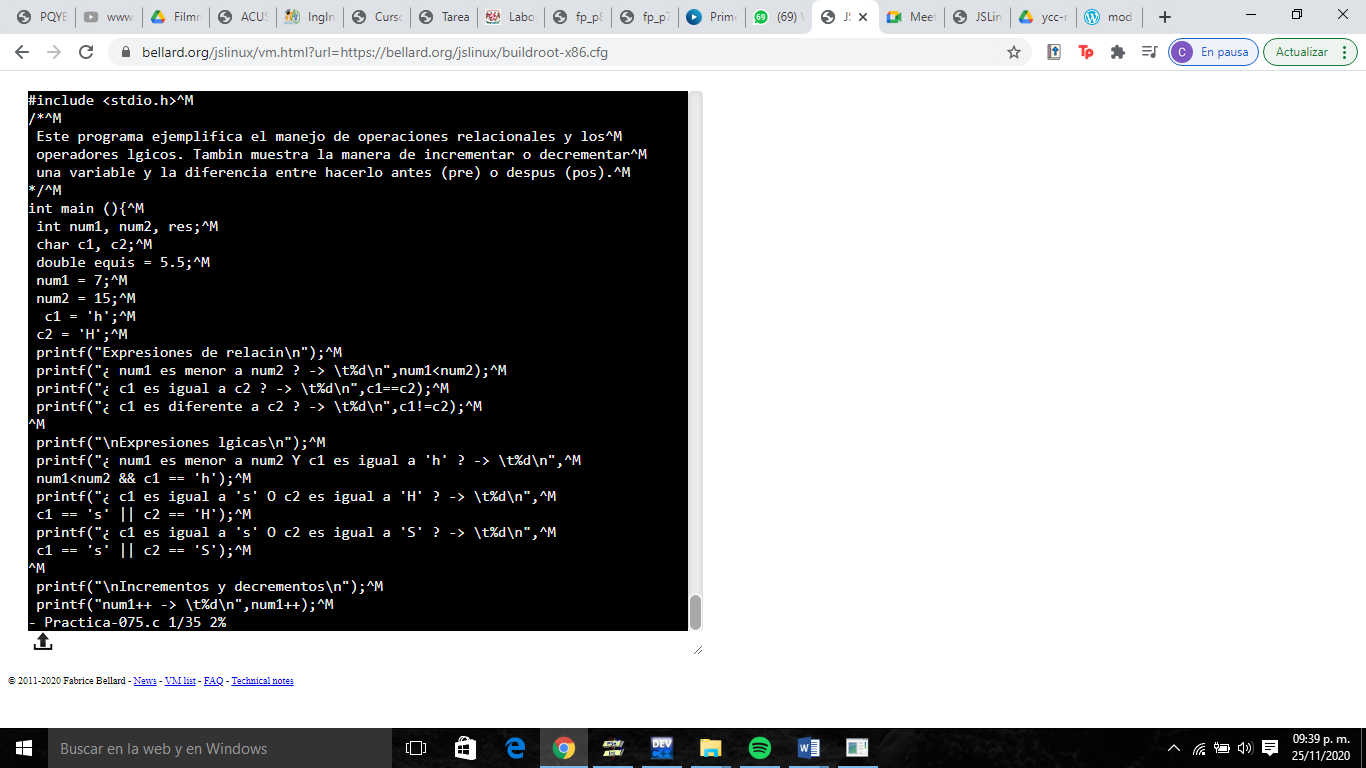
Agregamos un archivo



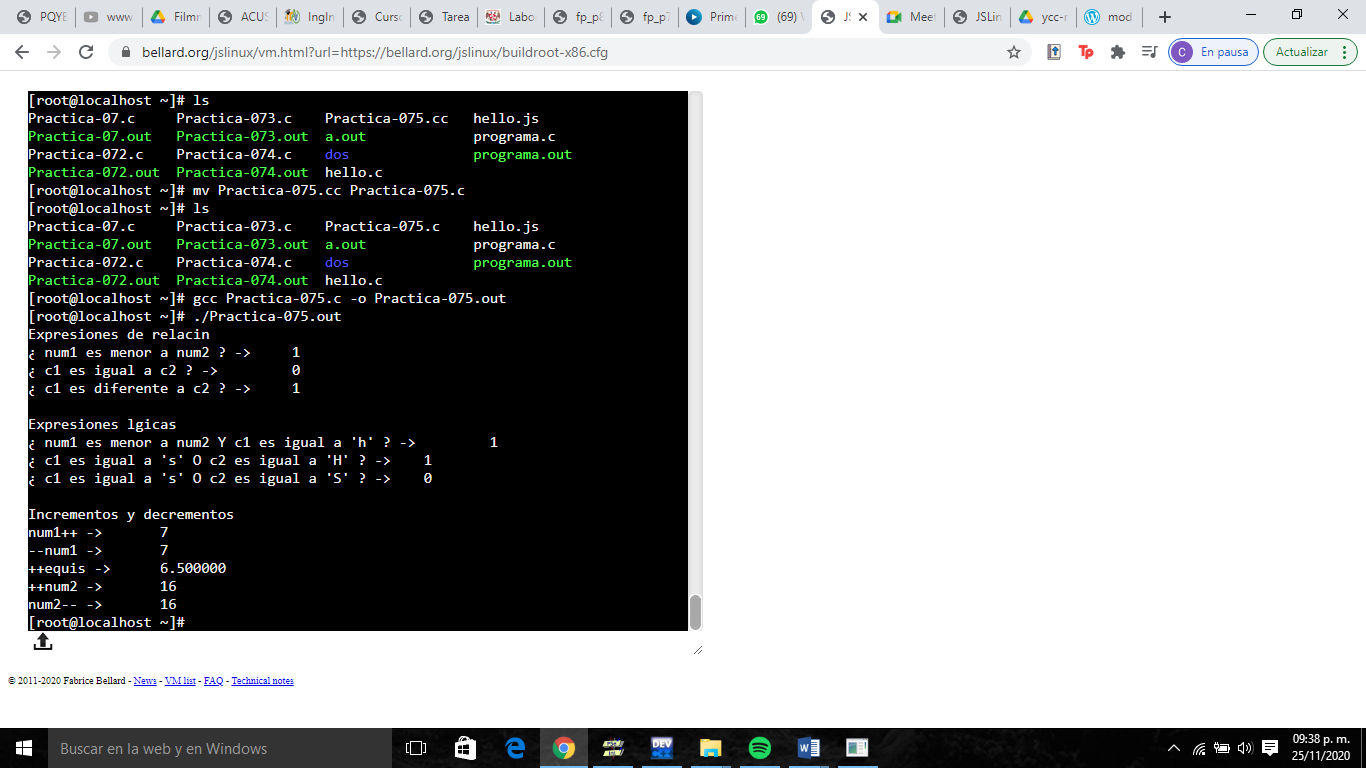
Compilamos y ejecutamos

-El valor estatico cambió y el valor contante no cambió

5.- Ejercicio



Modificamos y agregamos num2 en pre-incremento y pos-decremento



Compilamos y ejecutamos, nos muestra el resultado de num2 en pre-incremento y pos-decremento

**Tarea:**

1.- Investigar cual es el dato que se encuentra por default en en lenguaje C( signed o unsigned)

- Si se omite el clasificador por defecto se considera 'signed'.

2.- Indicar que sucede cuando en una variable tipo carácter se emplea el formato %d, %i, %o, %x

Cuando se trata de %d, %i, %o y %x se emplea Entero o Carácter

3.- Mencionar las características con las que debe crearse una variable

Debe iniciar con una letra [a-z].

Puede contener letras [A-Z, a-z], números [0-9] y el carácter guión bajo (\_).

4.- ¿Cuál es la diferencia entre variable estática y constante?

La variable estática tiene como característica poder cambiar de valor durante el proceso

La variable constante tiene como característica poder no cambiar de valor durante el proceso, su valor se queda en la memoria

5.- Menciona en que momento empleas los dos tipos de diferentes ( < >   != )

Se usa < para indicar Menor que. Ejemplo: a<5

Se usa > para indicar Mayor que. Ejemplo: a>5

Se usa != para indicar Diferente a. Ejemplo: ‘a’ != ‘b’

6.- Crea un programa en el que declares 4 variables haciendo uso de las reglas signed/unsigned, las cuatro variables deben ser solicitadas al usuario(se emplea scanf) y deben mostrarse en pantalla (emplear printf)

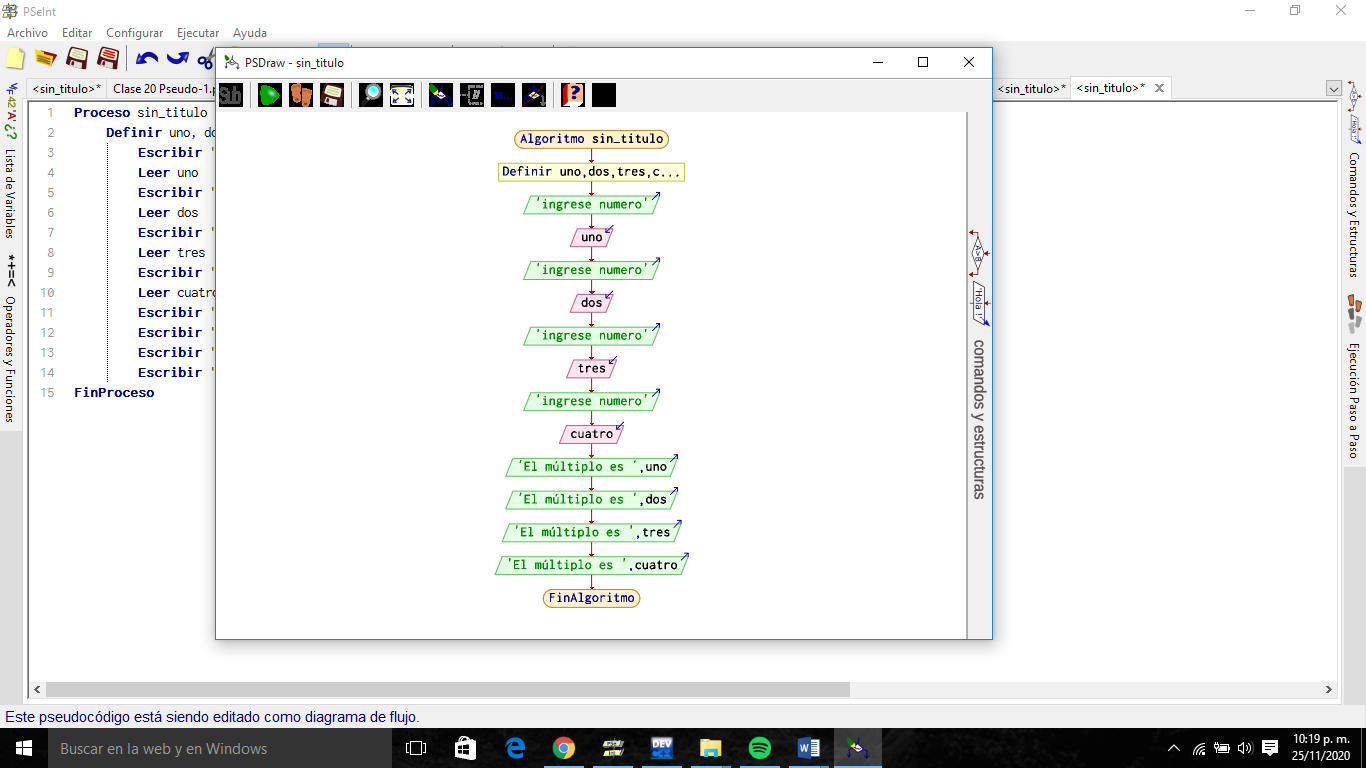
*Análisis*

Datos de entrada: uno, dos, tres, cuatro

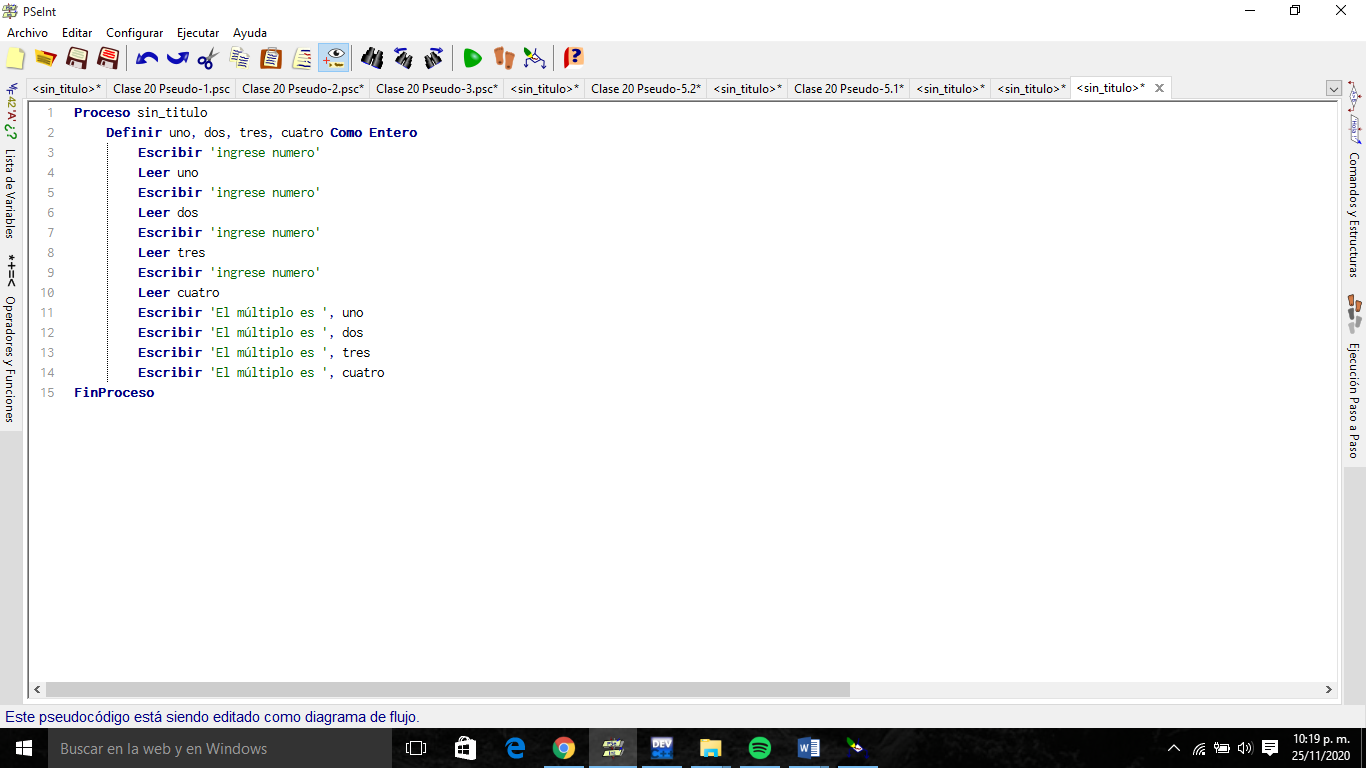
Restricciones: Solo enteros

Datos de salida: Imprimir los 4 valores

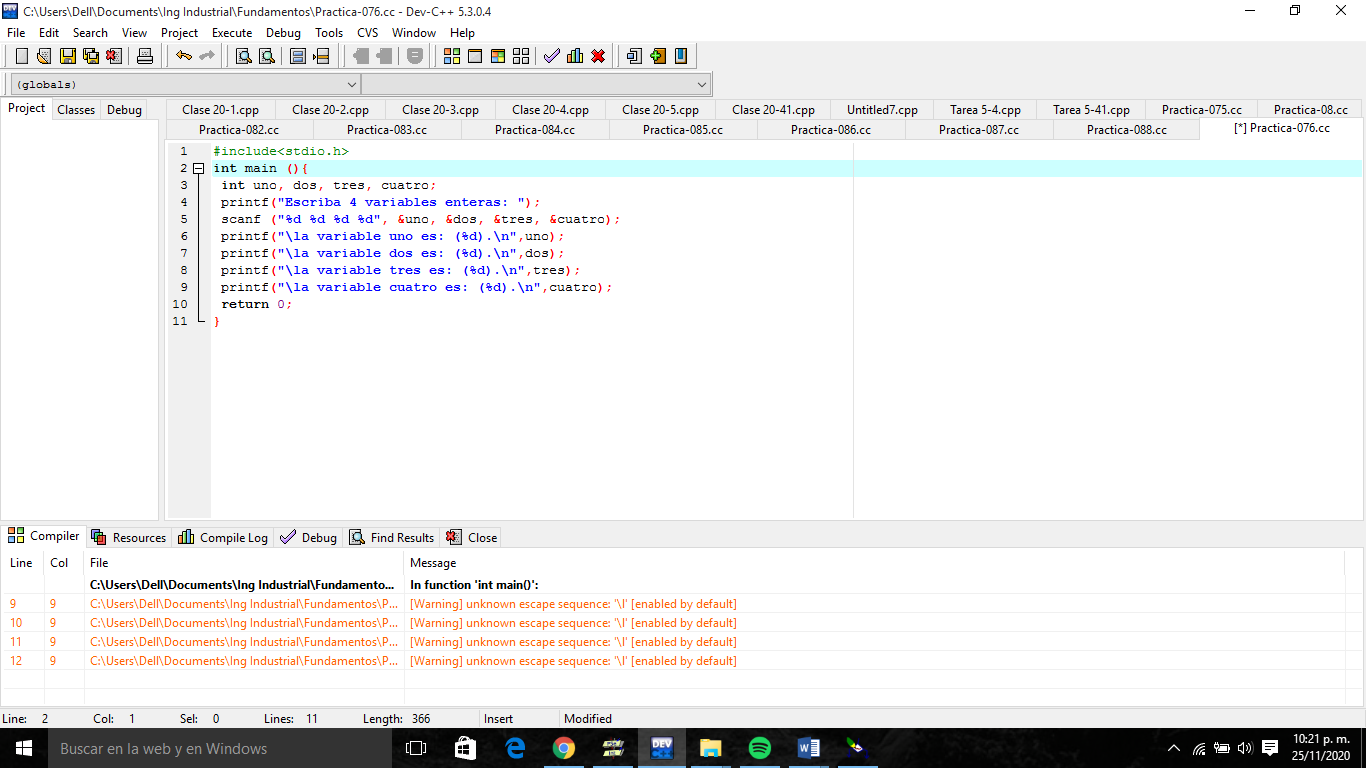
*Diagrama*

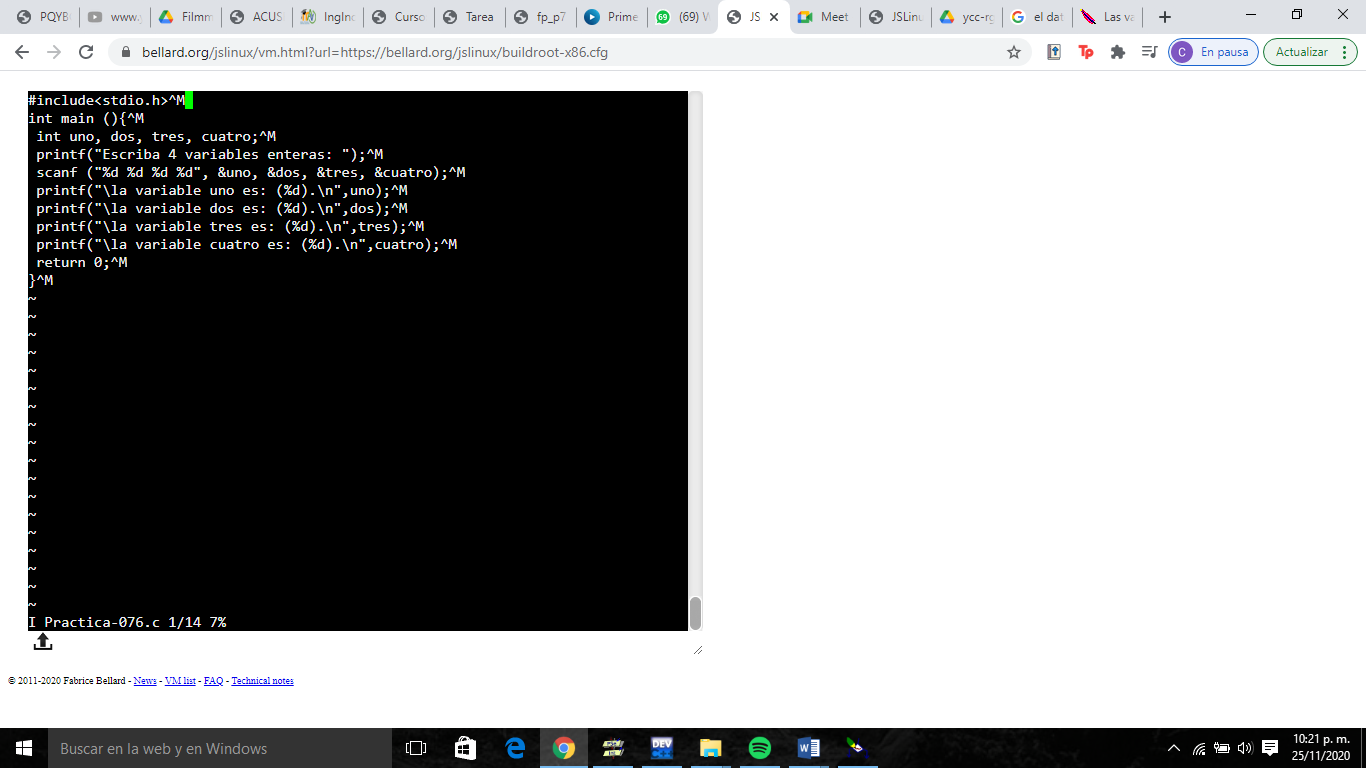


*Pseudocódigo*

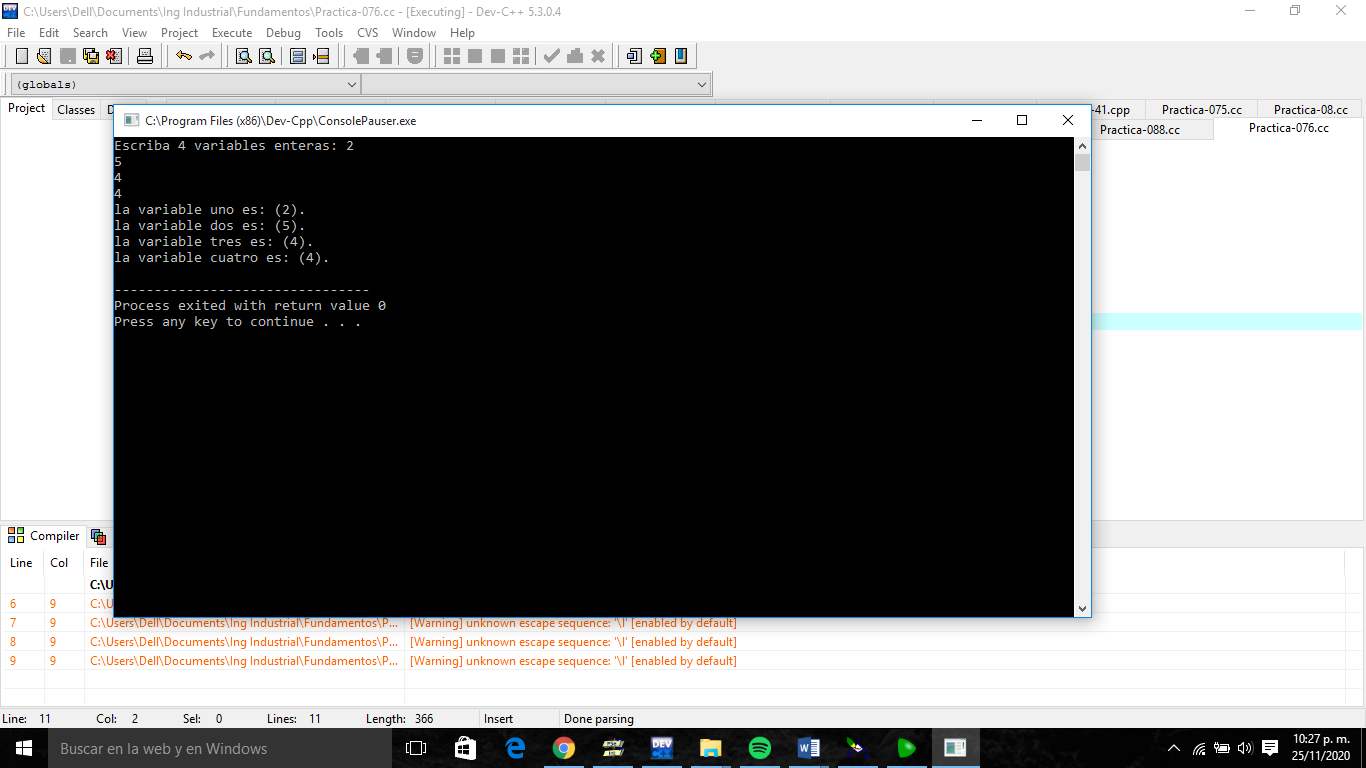
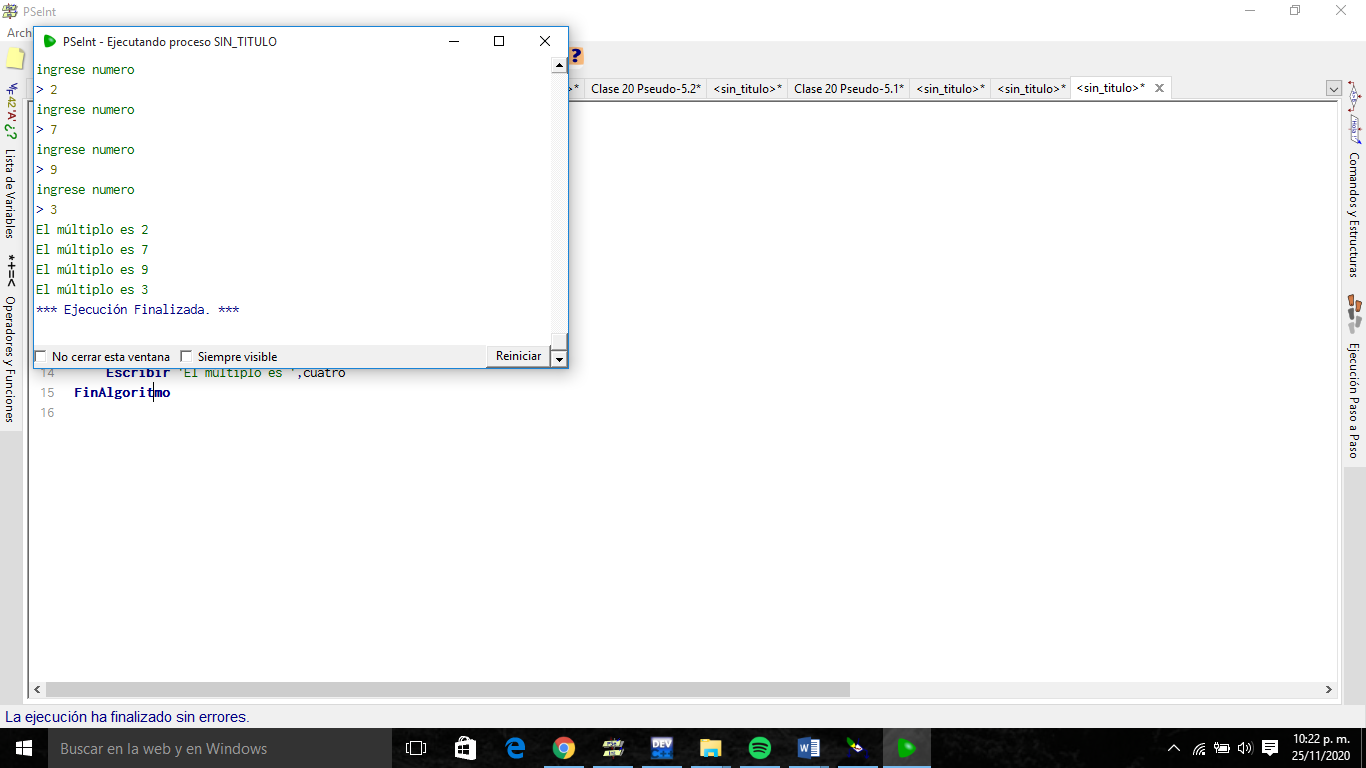


*Código*





*Pruebas*



7.- Crea un programa que le solicite su edad al usuario, leer el datos( emplear scanf) y mostrarlo en pantalla

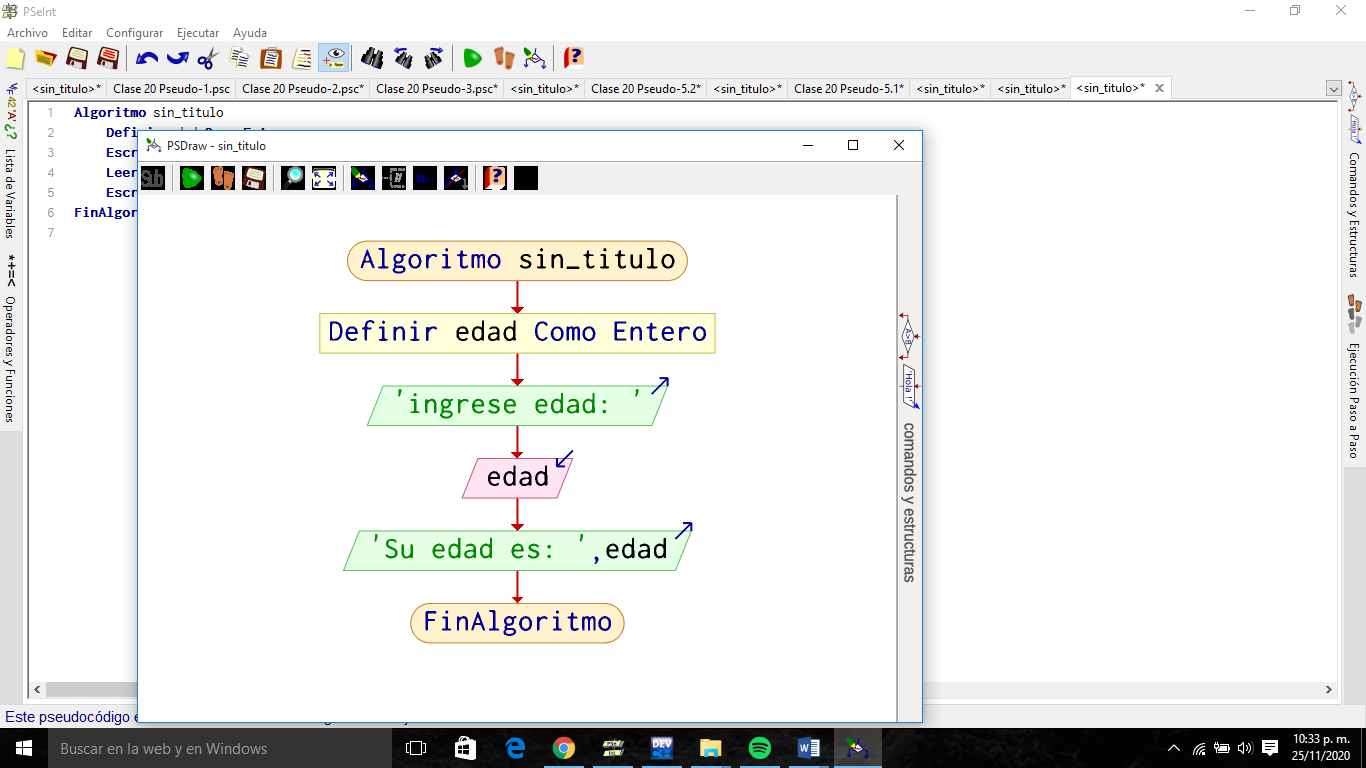
*Análisis*

Datos de entrada: edad

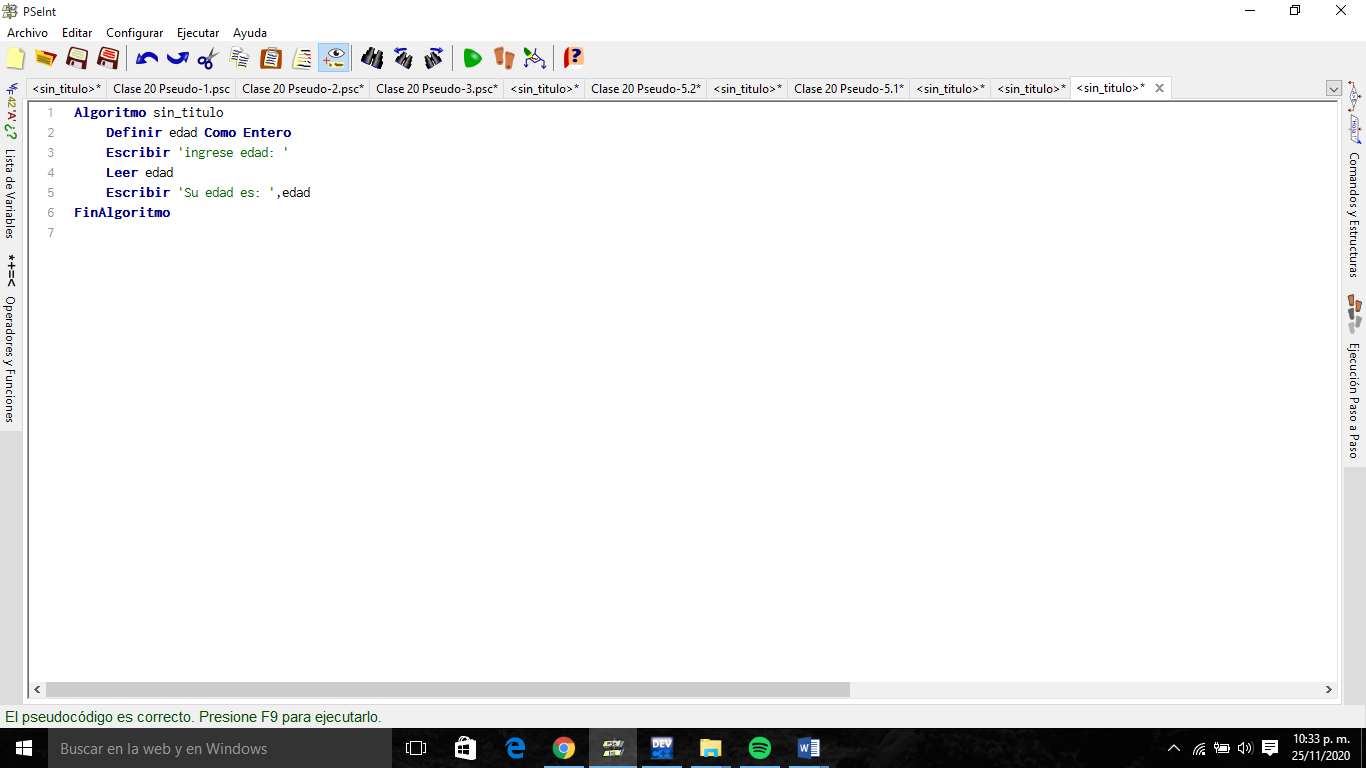
Restricciones: Solo enteros

Datos de salida: Imprimir la edad

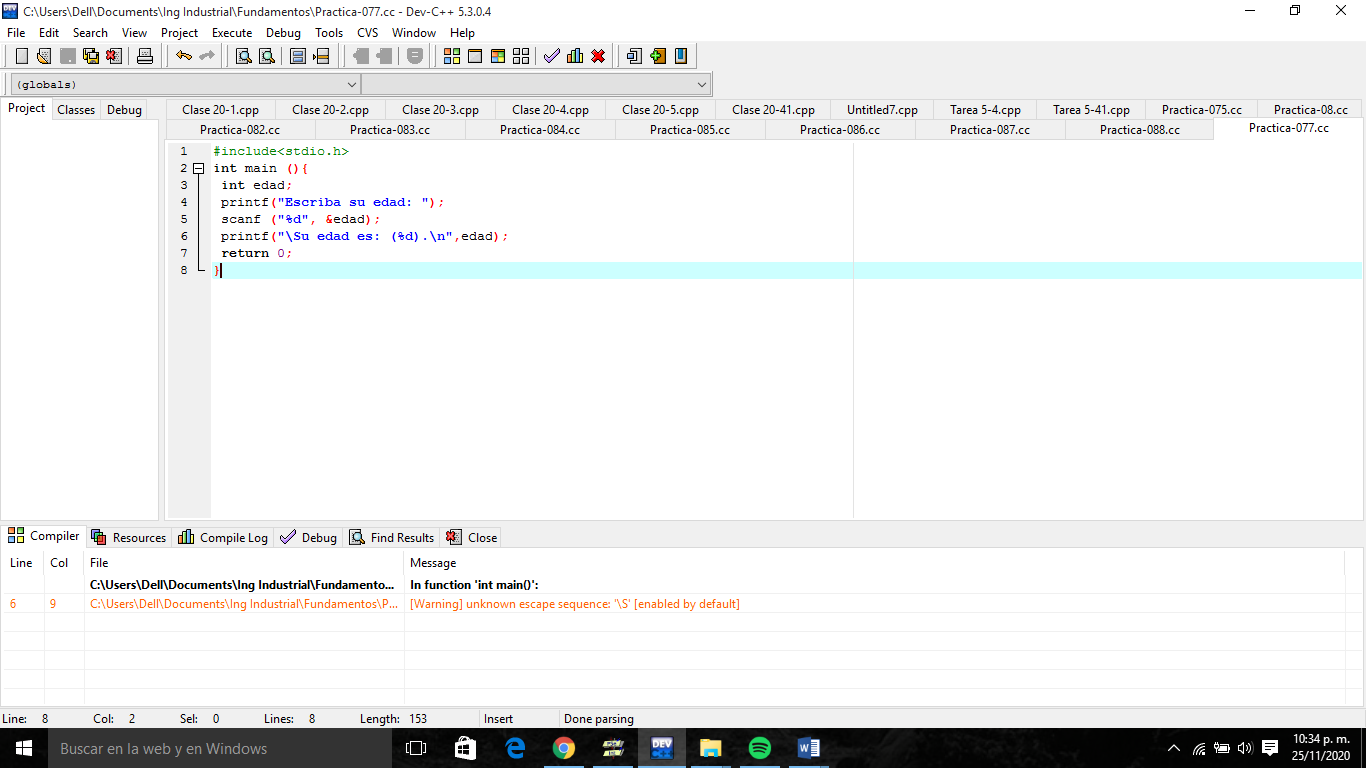
*Diagrama*

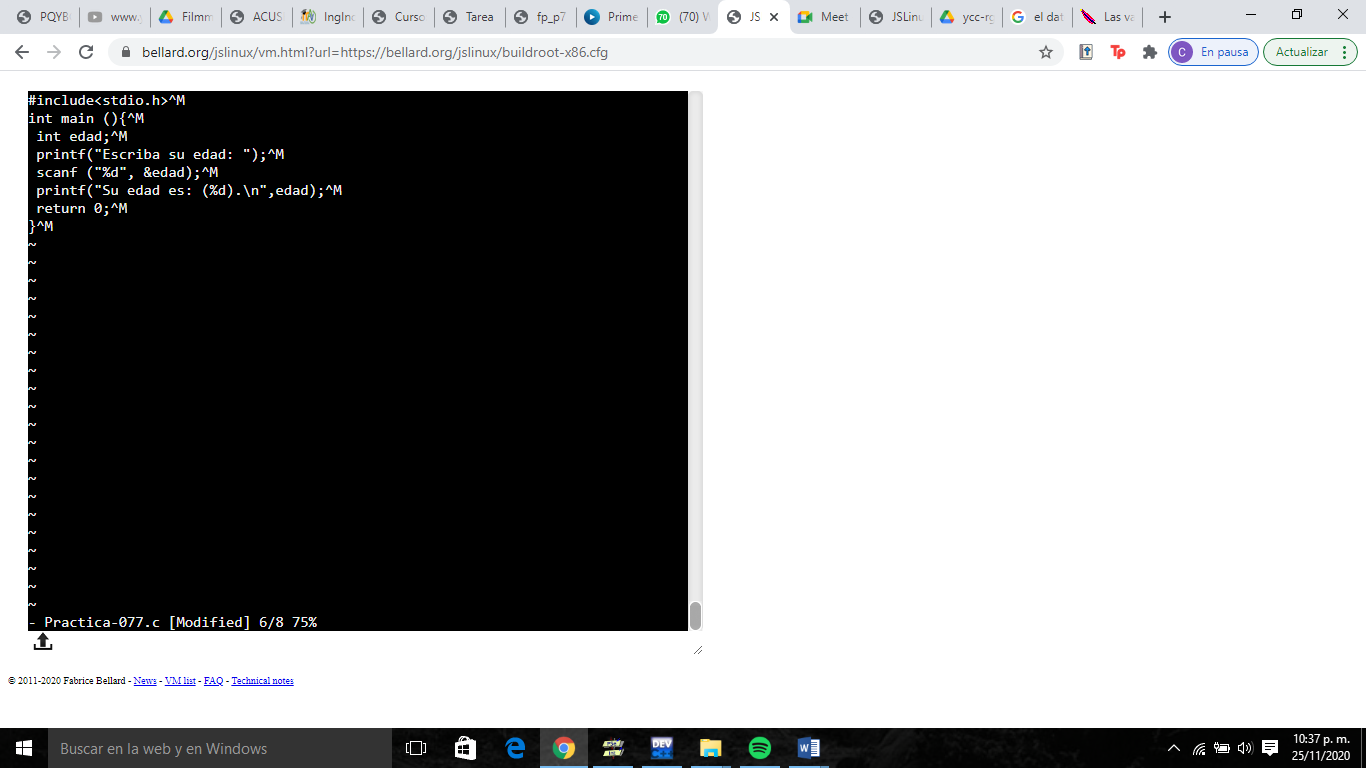


*Pseudocódigo*

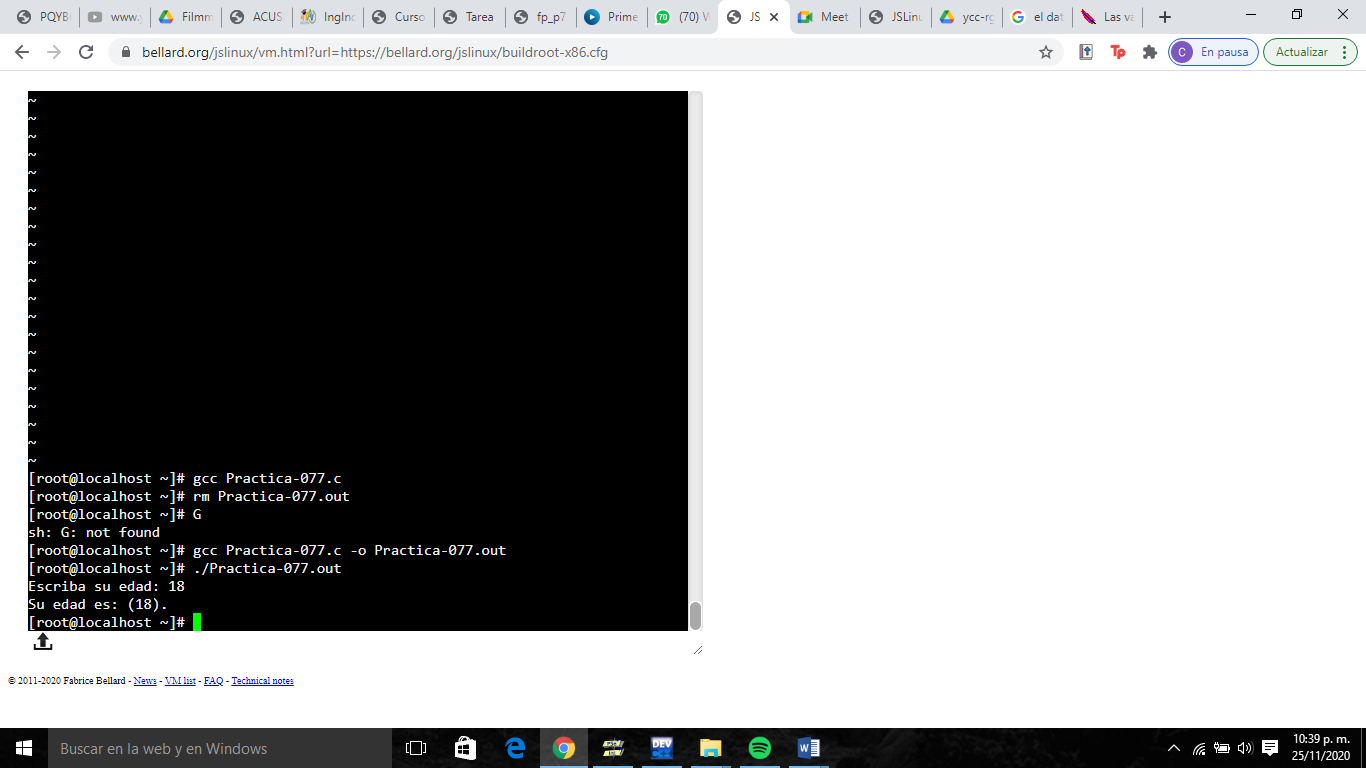


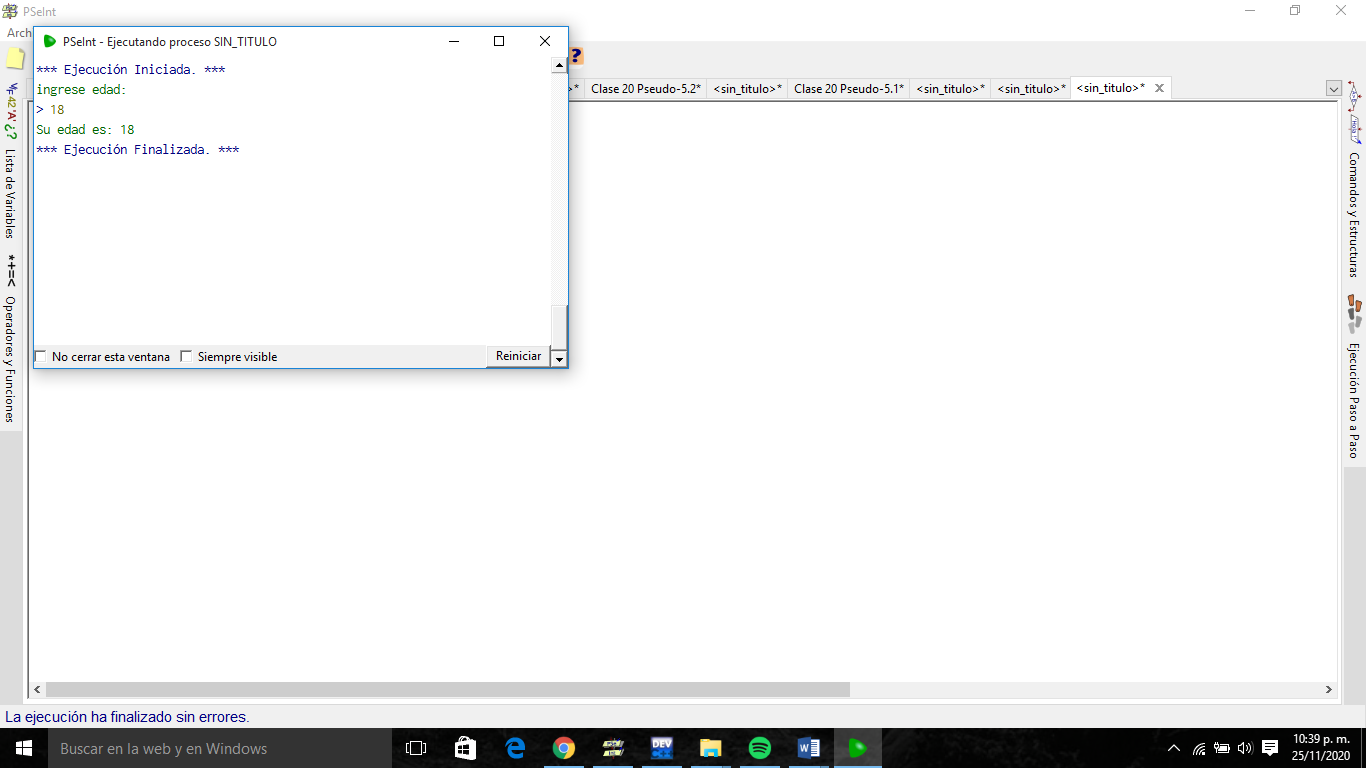
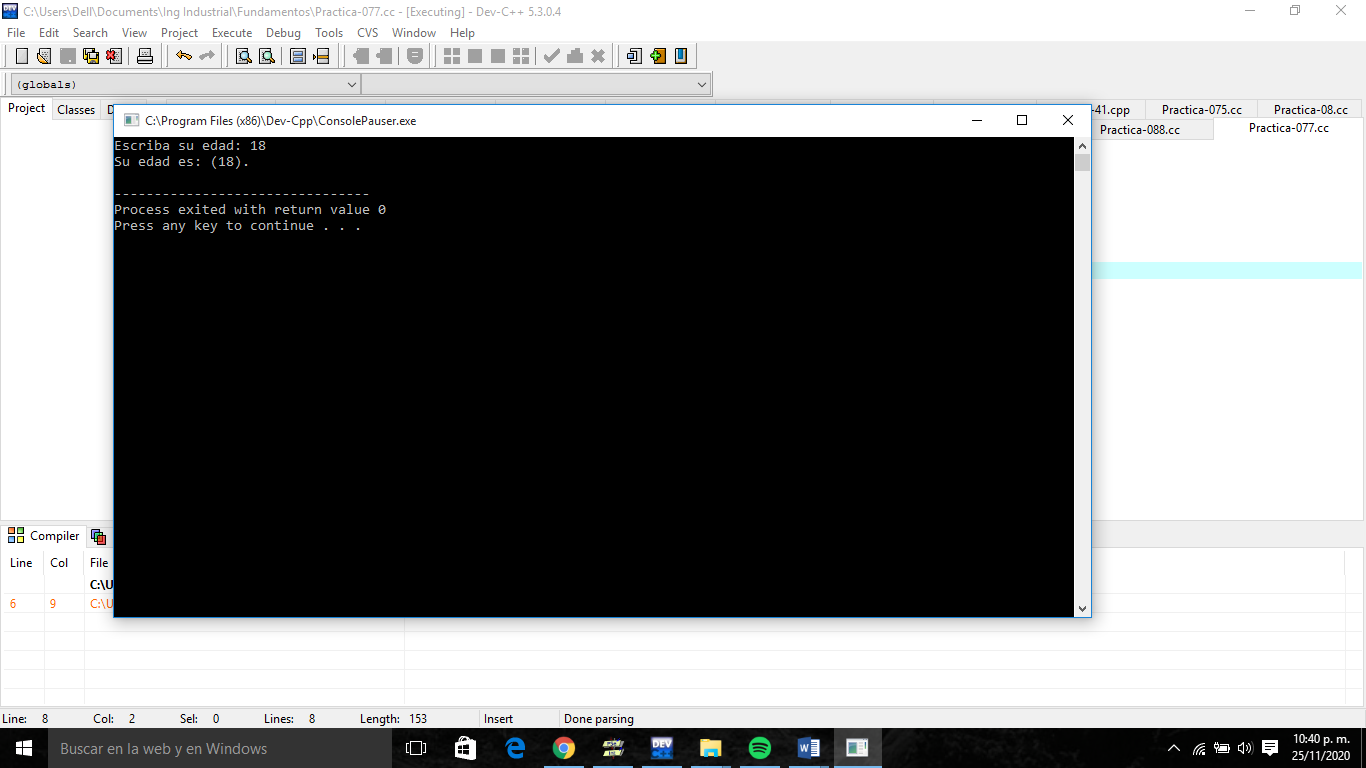
*Código*





*Pruebas*



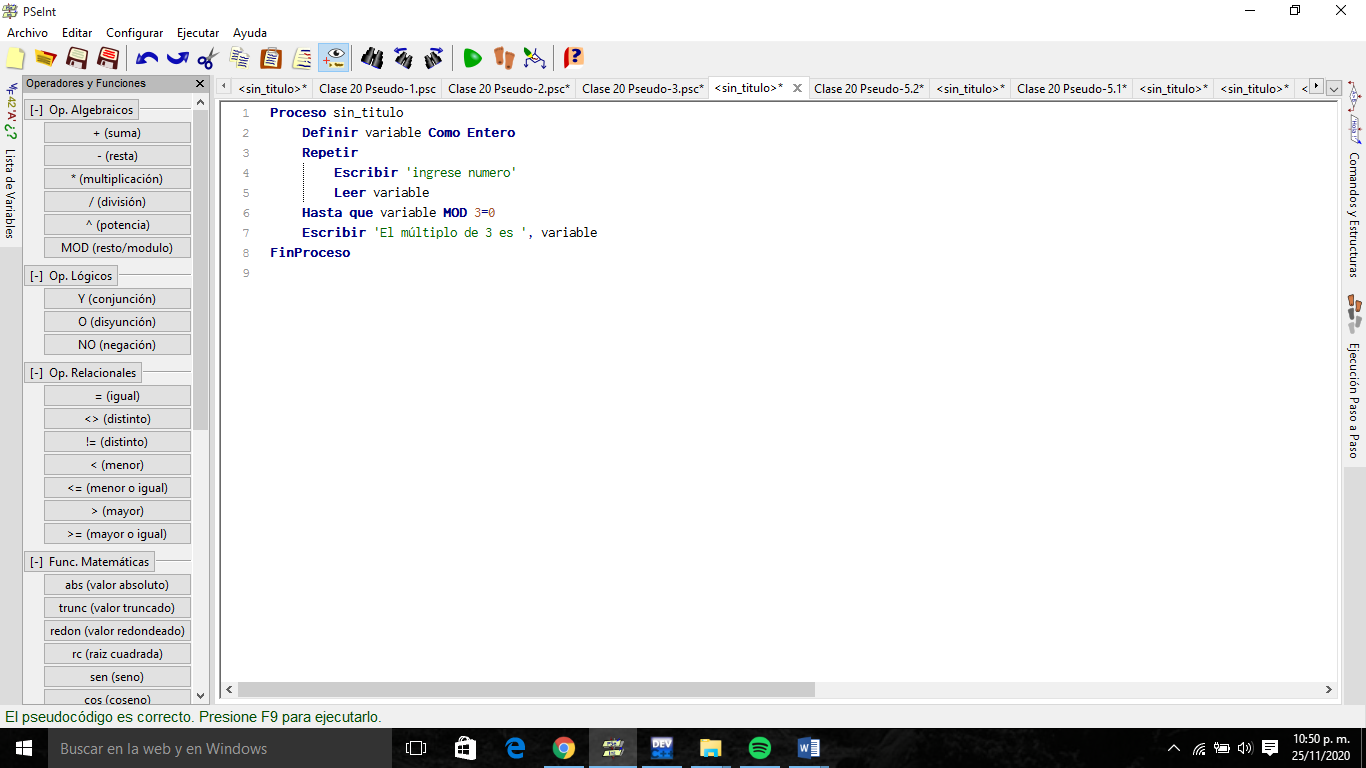
 

8.- Revisar y colocar cuando se emplea MOD y cuando se emplea % ([pseudocódigo](https://tuaulavirtual.educatic.unam.mx/mod/resource/view.php?id=167013" \o "Pseudocódigo) y códificación) agrega un ejemplo de su uso.

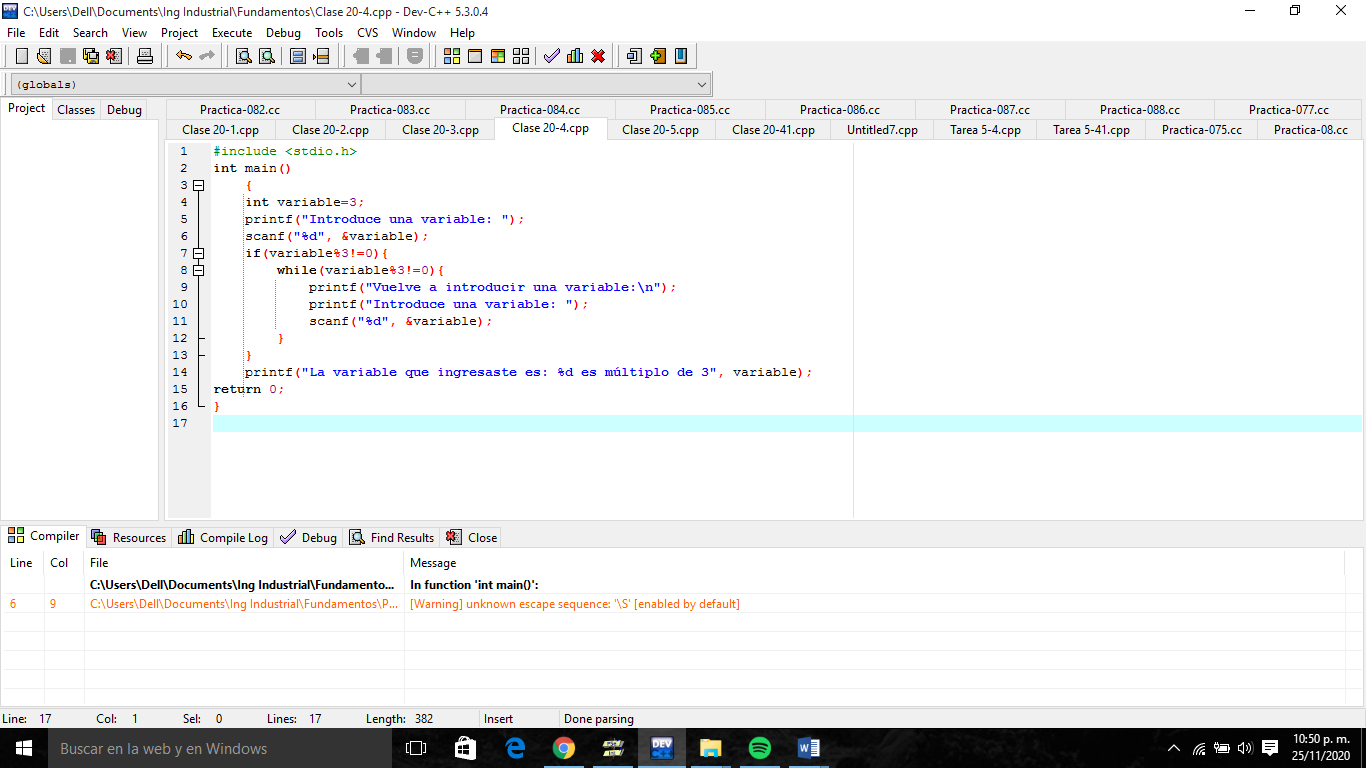
Se usa MOD cuando se necesita el resto de la acción anterior.

Ejemplo Hasta que variable MOD 3=0

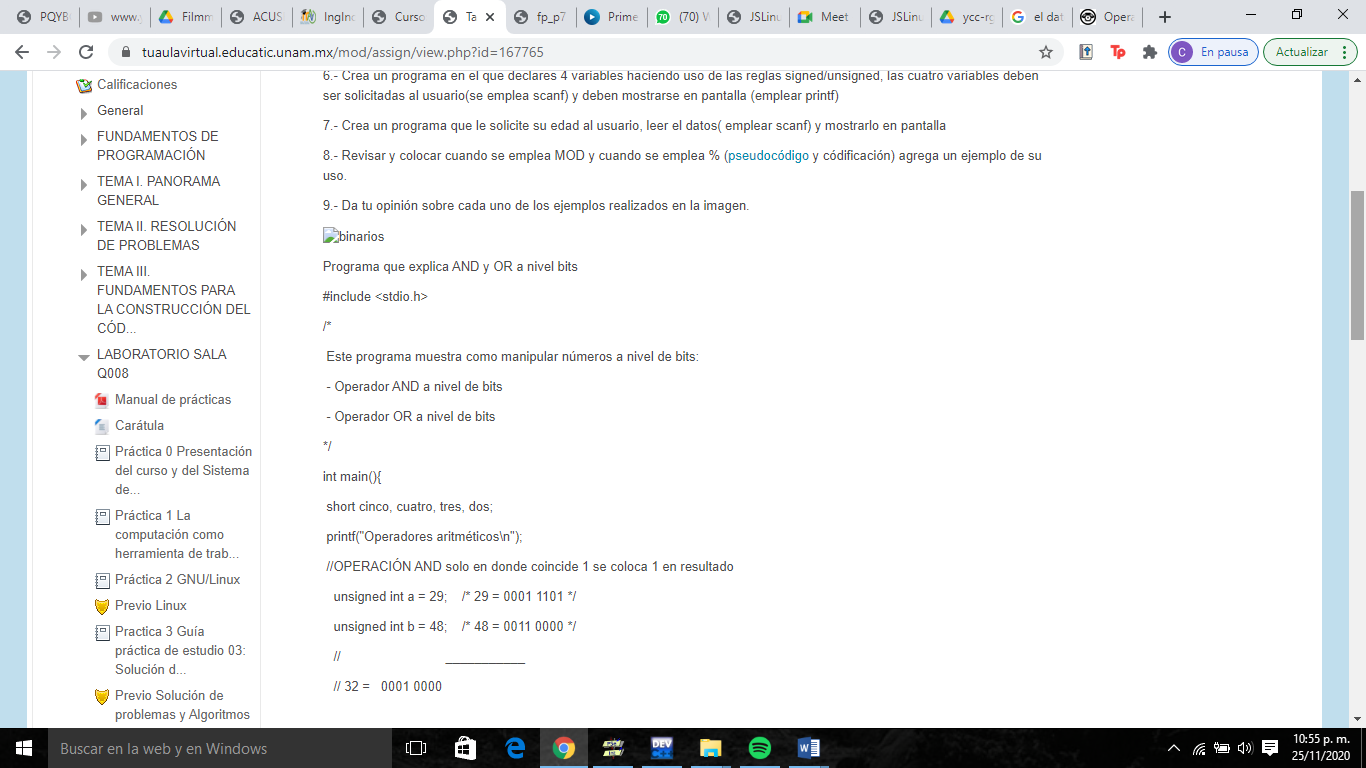
Pseudocódigo



Código



9.- Da tu opinión sobre cada uno de los ejemplos realizados en la imagen.



#include <stdio.h>

/\*

Este programa muestra como manipular números a nivel de bits:

- Operador AND a nivel de bits

- Operador OR a nivel de bits

\*/

int main(){

short cinco, cuatro, tres, dos;

printf("Operadores aritméticos\n");

//OPERACIÓN AND solo en donde coincide 1 se coloca 1 en resultado

unsigned int a = 29; /\* 29 = 0001 1101 \*/

unsigned int b = 48; /\* 48 = 0011 0000 \*/

// \_\_\_\_\_\_\_\_\_\_\_

// 32 = 0001 0000

cinco = 5; //5 en binario: 0000 0000 0000 0101

cuatro = 4;//4 en binario: 0000 0000 0000 0100

// \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

// 4= 0000 0000 0000 0100

printf("29 & 48 = %d\n", a&b );

printf("5 & 4 = %d\n",cinco&cuatro);

printf("4 & 3 = %d\n",cuatro&tres);

//OPERACIÓN OR todos los 1 se colocan en el resultado

tres = 3;// 3 en binario: 0000 0000 0000 0011

dos = 2; /\* 2 en binario: 0000 0000 0000 0010

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

3= 0000 0000 0000 0011

5 en binario: 0000 0000 0000 0101

4 en binario: 0000 0000 0000 0100

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

5= 0000 0000 0000 0101

29 = 0001 1101

48 = 0011 0000

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

61 = 0011 1101 \*/

printf("3 | 2 = %d\n",tres|dos);

printf("5 | 4 = %d\n",cinco|cuatro);

printf("29 | 48 = %d\n", a|b );

printf("\n");

return 0;

}

En mi opinión es un programa que tiene como objetivo hacer operaciones mediante el uso de su parte binaria, lo cual es interesante ya que es otra forma de obtener resultados de sumas, restas de binarios

**Conclusiones:**

En esta ocasión elaboramos más funciones en el programa Linux; muy similar a llevar acabo códigos en DEV, nuevamente las actividades que le llevaron a cabo sirven para su mejor y más ágil entendimiento del programa, usando de alguna manera “nuevas” funciones para la realización de nuestro código, y aún más sobre el análisis de problemas, conociendo varias formas de resolverlo, en esta ocasión la elaboración de un código demás de DEV.

**Conclusiones:**

* El lenguaje de programación C. Brian W. Kernighan, Dennis M. Ritchie, segunda edición, USA, Pearson Educación 1991.
* Carlos Guadalupe (2013). Aseguramiento de la calidad del software (SQA). [Figura 1]. Consulta: Junio de 2015. Disponible en: <https://www.mindmeister.com/es/273953719/aseguramiento-de-la-calidad-delsoftware-sqa>
* Laboratorio Salas A y B. (2015). Retrieved November 4, 2020, from Unam.mx website: <http://lcp02.fi-b.unam.mx/>