|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **NOMBRE DE LA PRÁCTICA** | **INTRODUCCIÓN AL LENGUAJE C.** | | | **NO.** | **1** |
| **ASIGNATURA:** | **MÉTODOS NUMÉRICOS.** | **CARRERA:** | **INGENIERÍA EN**  **SISTEMAS**  **COMPUTACIONALES** | **DURACIÓN**  **DE LA**  **PRÁCTICA**  **(HRS)** | **2**  **HORAS** |

**NOMBRE DEL ALUMNO: Jose Manuel Arce Salvador**

**GRUPO: 3401**

1. **Competencia(s) específica(s):** 
   * Aplica los tipos de errores para identificar la incertidumbre y limitaciones de los cálculos numéricos en una computadora.

Encuadre con CACEI: Registra el (los) atributo(s) de egreso y los criterios de desempeño que se evaluarán en esta práctica.

|  |  |
| --- | --- |
| **No. atributo** | **Atributos de egreso del PE que impactan en la asignatura** |
| **1** | **Reconocer la necesidad permanente de**  **conocimiento adicional y tener la habilidad para localizar, evaluar, integrar y aplicar este conocimiento adecuadamente.** |

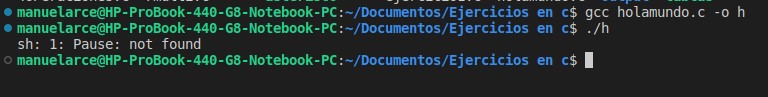
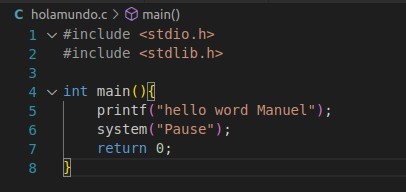
1. **Lugar de realización de la práctica (laboratorio, taller, aula u otro):**  Aula de clases.

1. **Material empleado:** 
   * Computadora personal.
   * SO Linux.
   * Dev C++.
2. **Desarrollo de la práctica:**

**EJERCICIOS ELABORADOS EN CLASE**

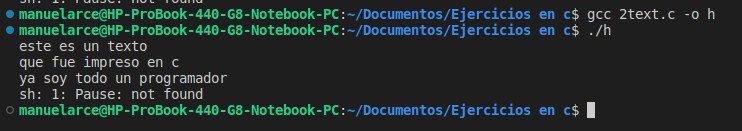
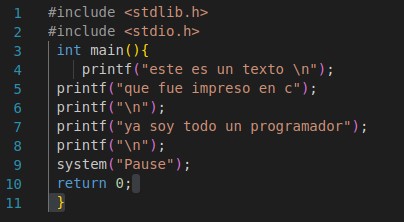
# Ejercicio 1 “Hola mundo”

En este primer programa codificamos nuestro primer “Hola Mundo” en C++, para lo cual aprendimos algunas reglas de sintaxis para hacer posible el correcto funcionamiento de nuestro programa.



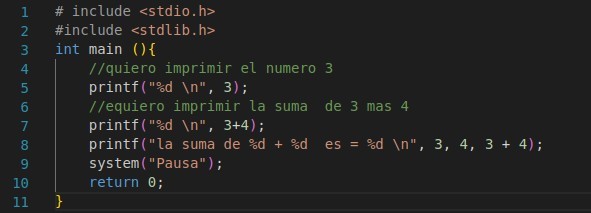
# Ejercicio 2 “Segundo Texto”

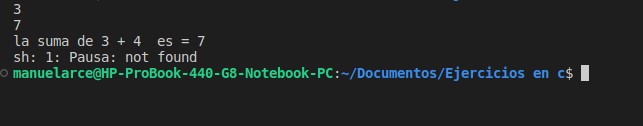
En este programa codificamos nuestro segundo texto en C++, para lo cual aprendimos algunas reglas de sintaxis para hacer posible el correcto funcionamiento de nuestro programa.



# Ejercicio 3 “Primera Suma”

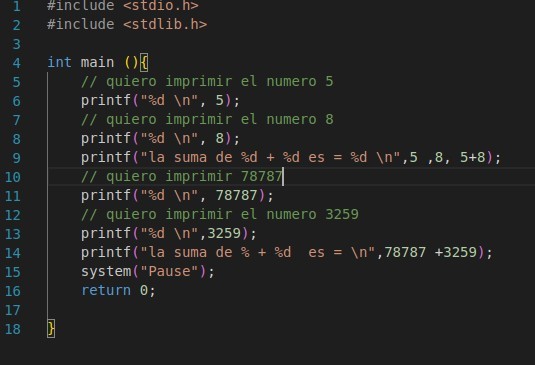
En este programa codificamos nuestra primera suma decimal en C++, para lo cual aprendimos algunas reglas de sintaxis numérica para hacer posible el funcionamiento de dicho programa.

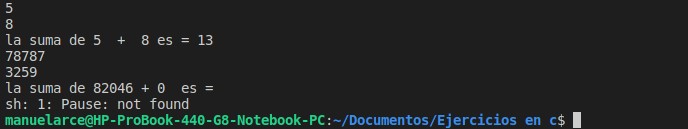




# Ejercicio 4 “Operaciones Aritméticas”

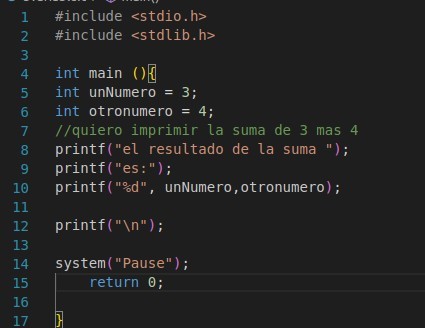
En este programa codificamos 2 operaciones aritméticas en C++, para lo cual aplicamos algunas reglas de sintaxis numérica para hacer posible el funcionamiento de dicho programa.

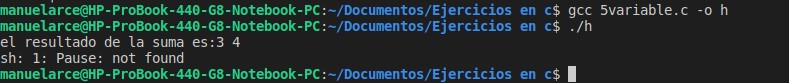




# Ejercicio 5 “Ejemplo declaración de variables”

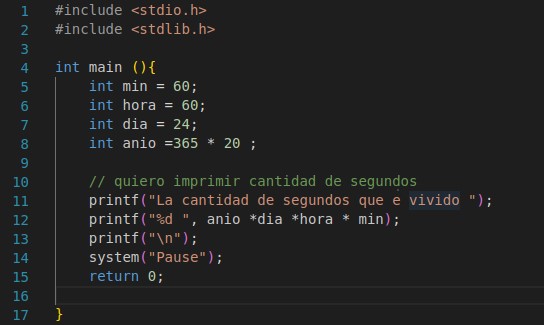
En este programa codificamos 2 variables aritméticas con un valor definido en C++ de tipo entero, para lo cual aplicamos algunas reglas de sintaxis numérica para hacer posible el funcionamiento de dicho programa.

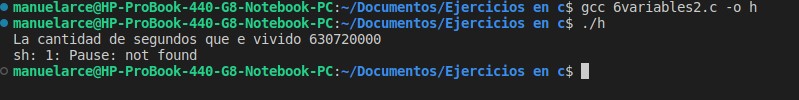




# Ejercicio 6 ”Segundos de vida”

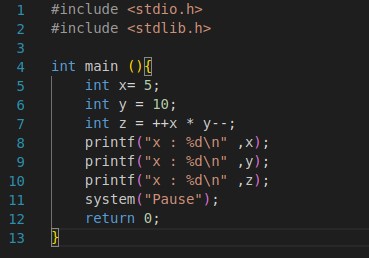
En este programa codificamos nuestras variables aritméticas con un valor definido en C++ de tipo entero, para lo cual aplicamos algunas reglas de sintaxis numérica y de operaciones, para obtener los segundos de vida.

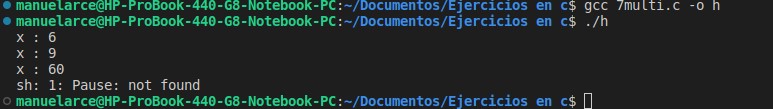




# Ejercicio 7 ”Multiplicación”

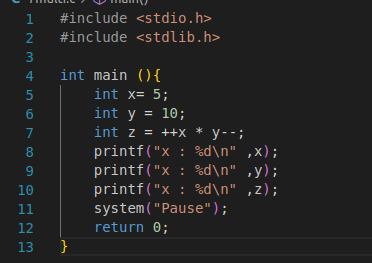
En este programa codificamos nuestras variables aritméticas con un valor definido en C++ de tipo entero, para lo cual aplicamos algunas reglas de sintaxis numérica y de operaciones, para obtener el resultado de nuestra multiplicación**.**

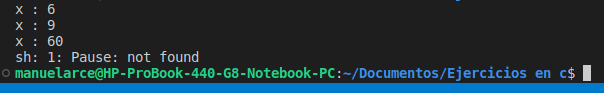




# Ejercicio 8 ”Operadores”

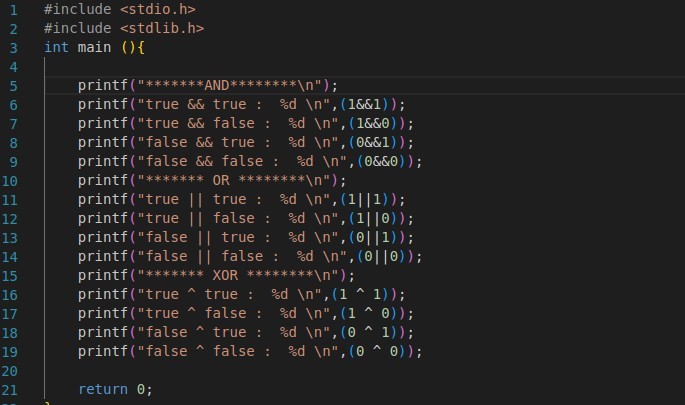
En este programa codificamos nuestras variables aritméticas con un valor definido en C++ de tipo entero, así como también algunos operadores aritméticos a usar.

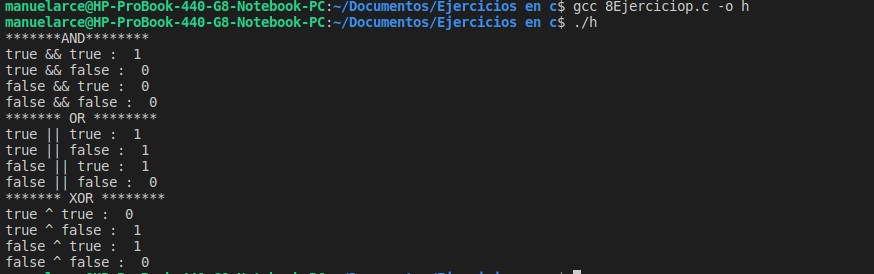


****

# Ejercicio 9 ”Operadores”

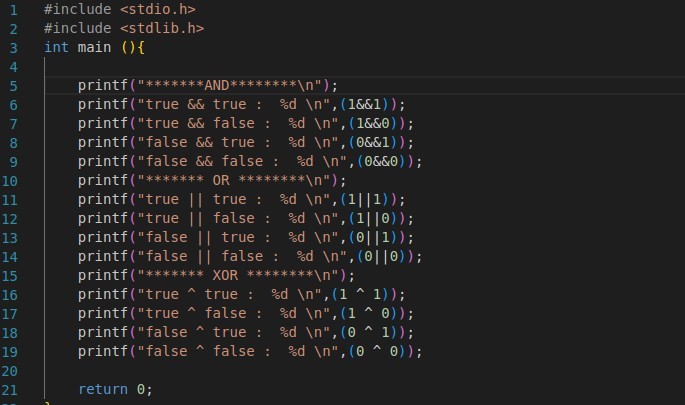
En este programa codificamos nuestras variables aritméticas con un valor definido en C++ de tipo entero, así como también algunos operadores aritméticos a usar para resolver las tablas de verdad.





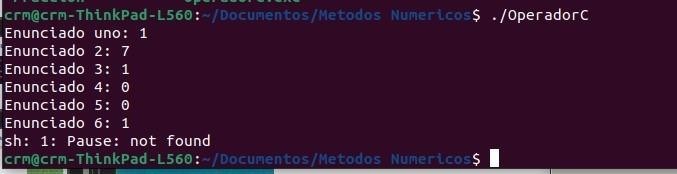
# Ejercicio 10 ”Operadores”

En este programa codificamos nuestras variables aritméticas con un valor definido en C++ de tipo entero, así como también algunos operadores aritméticos a usar para resolver las tablas de verdad.



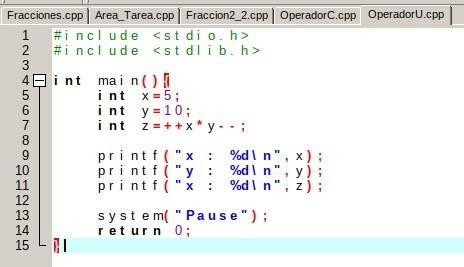
# Ejercicio 11” Operadores Lógicos”

En este programa codificamos nuestras variables aritméticas con un valor definido en C++ de tipo double, así como también algunos operadores con el fin de resolver un problema con lógico planteado.



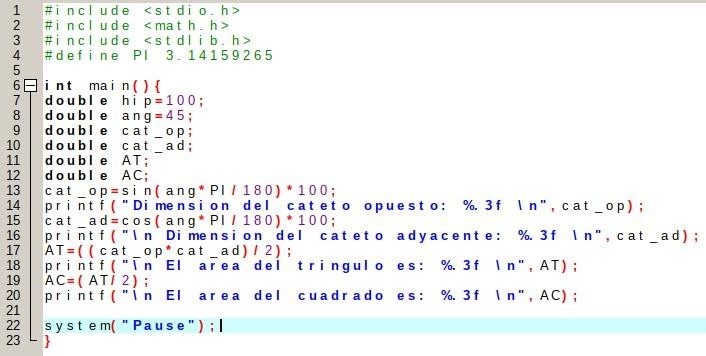
# Ejercicio 12” Operadores Lógicos”

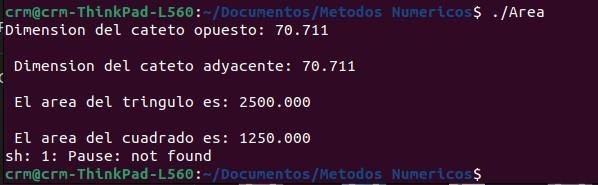
En este programa codificamos nuestras variables aritméticas con un valor definido en C++ de tipo double, así como también algunos operadores con el fin de resolver un problema con lógico planteado.



# Ejercicio 13” Área del Triángulo”

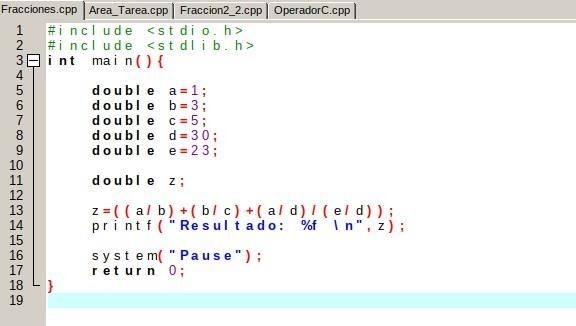
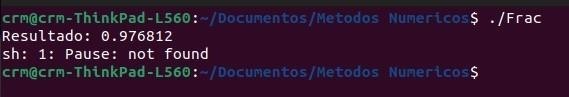
En este programa codificamos nuestras variables aritméticas con un valor definido en C++ de tipo double, así como también algunos operadores aritméticos y funciones de seno y coseno a usar para resolver un problema con el teorema de pitagoras.





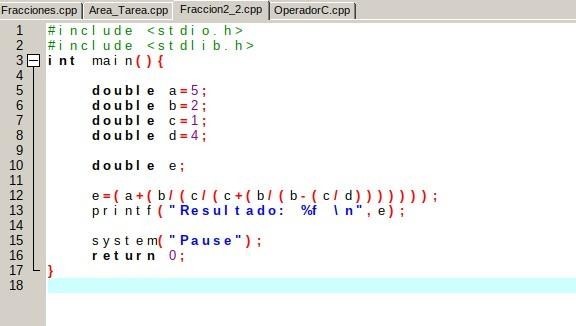
# Ejercicio 14 ”Fracción 1”

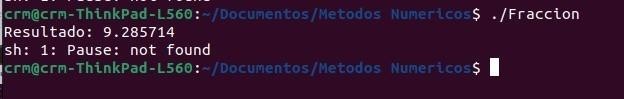
En este programa codificamos nuestras variables aritméticas con un valor definido en C++ de tipo double, así como también algunos operadores aritméticos ordenados para resolver un problema de acuerdo a la prioridad de operadores.



# Ejercicio 15 ”Fracción 2”

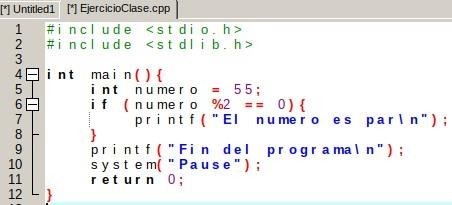
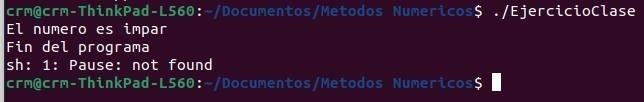
En este programa codificamos nuestras variables aritméticas con un valor definido en C++ de tipo double, así como también algunos operadores aritméticos ordenados para resolver un problema de acuerdo a la prioridad de operadores.





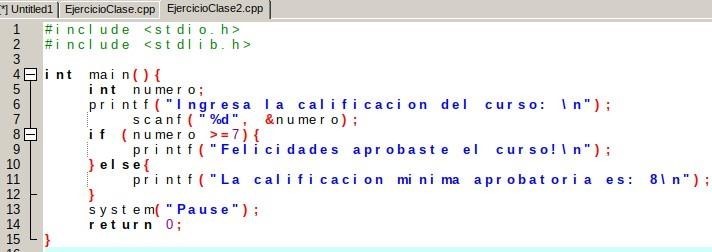
# Ejercicio 16 ”Numero par e impar”

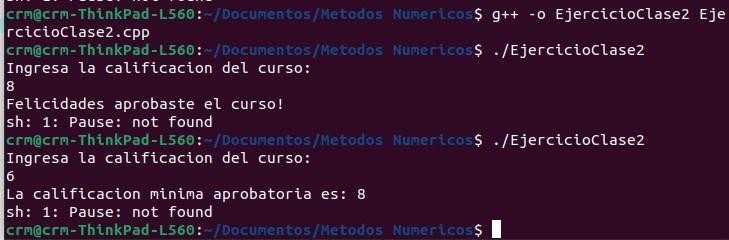
En este programa codificamos una variable del tipo entero con un valor definido y posteriormente con el uso de la condicional “if” evaluamos si este es par.



# Ejercicio 17 ”Evaluación y clasificación de un curso”

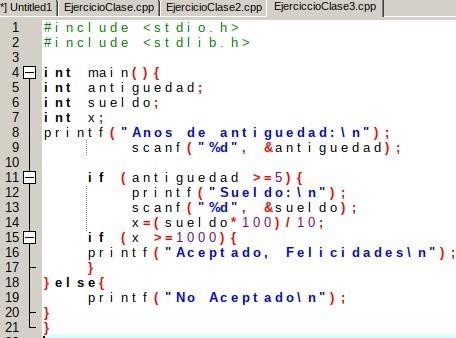
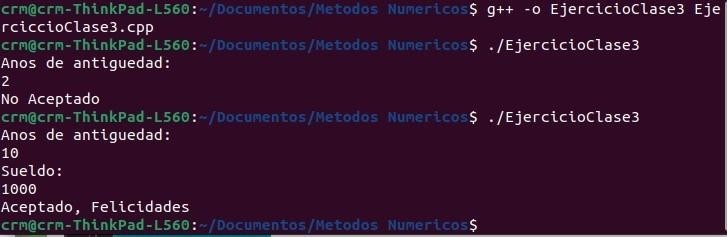
En este programa codificamos una variable con ingreso desde teclado que representa una calificación y posteriormente con el uso de la condicional “if” evaluamos si esta es óptima o no para aprobar un curso.





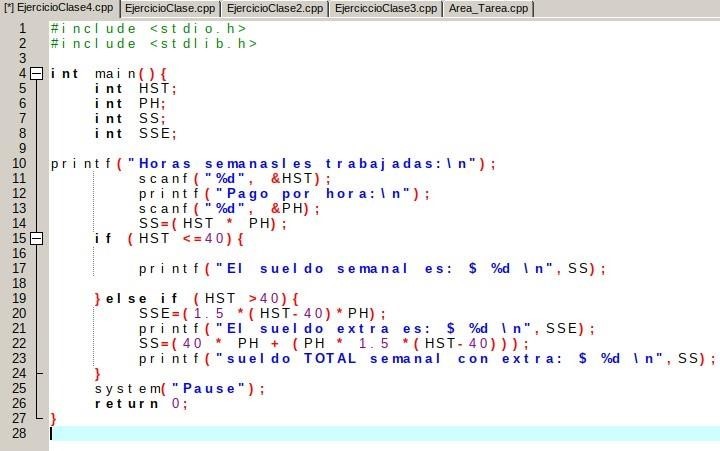
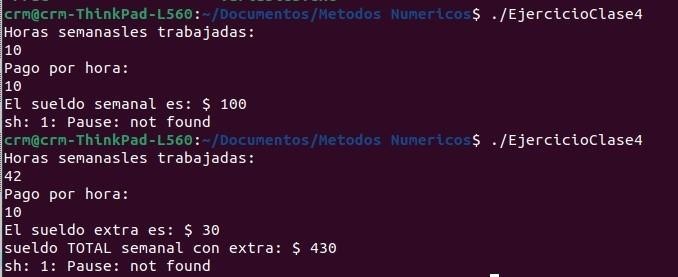
# Ejercicio 18” Evaluación antigüedad”

En este programa codificamos dos variables con ingreso desde teclado que representa la antigüedad y sueldo de un empleo para adquirir un crédito hipotecario y posteriormente con el uso de la condicional “if” evaluamos si es aceptado o no el usuario.



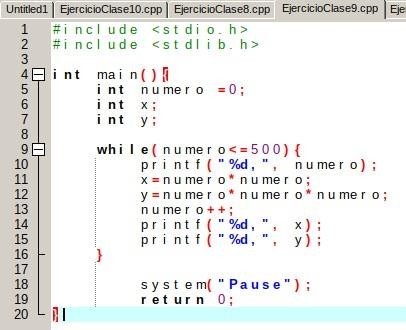
# Ejercicio 19 ”Horas de trabajo”

Realizar el cálculo el salario semanal de un empleado, sabiendo que éste se calcula en base a las horas semanales trabajadas y de acuerdo a un precio especificado por hora. Si se pasa de las 40 horas semanales, las horas extras se pagarán a razón de 1.5 veces la hora ordinaria.



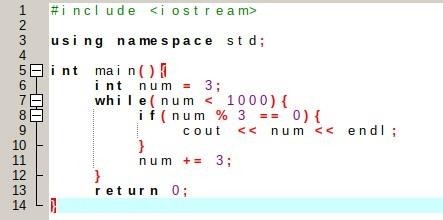
# Ejercicio 20 ”Cuadrados y Cubos”

Imprimir los cuadrados y los cubos de los primeros quinientos números naturales mediante la implementación del ciclo “while” que facilitara la codificación del código.



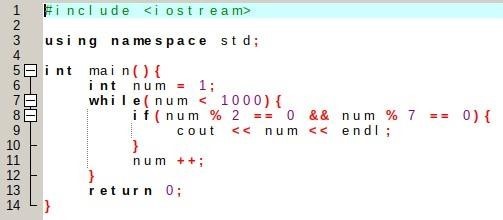
# Ejercicio 21 “Números divisibles”

Imprimir todos los números divisibles entre 3 mayores a cero y menores a mil usando la estructura del ciclo while.



# Ejercicio 22 “Números divisibles\_2”

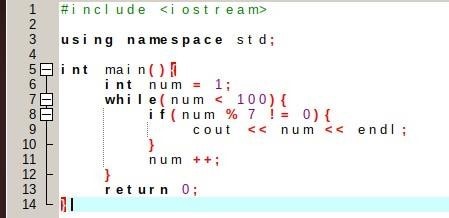
Imprimir todos los números que son divisibles entre 2 y entre 7, mayores a 0 y menores a mil, usando el ciclo “while”.





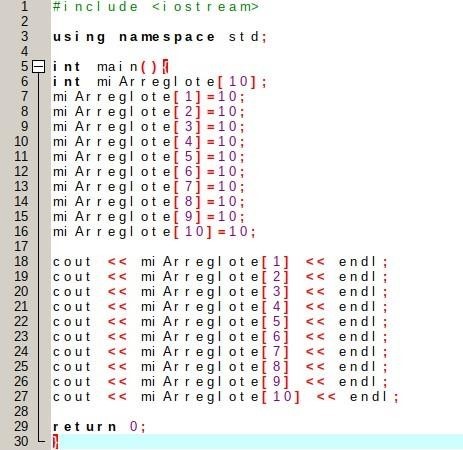
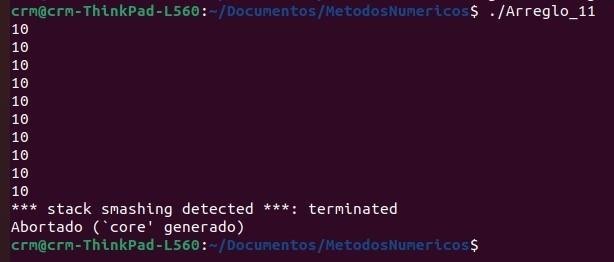
# Ejercicio 23 “Enteros Positivos < 100”

Escribir todos los enteros positivos menores que 100 omitiendo aquellos que son divisibles por 7 usando el ciclo “while”.



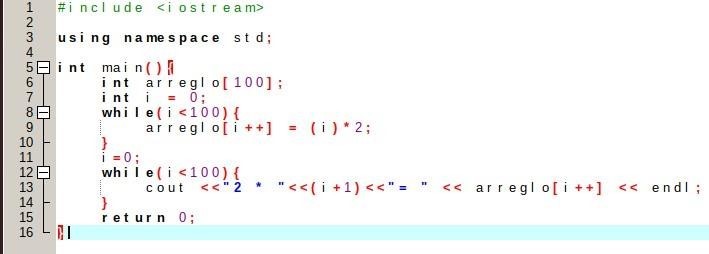
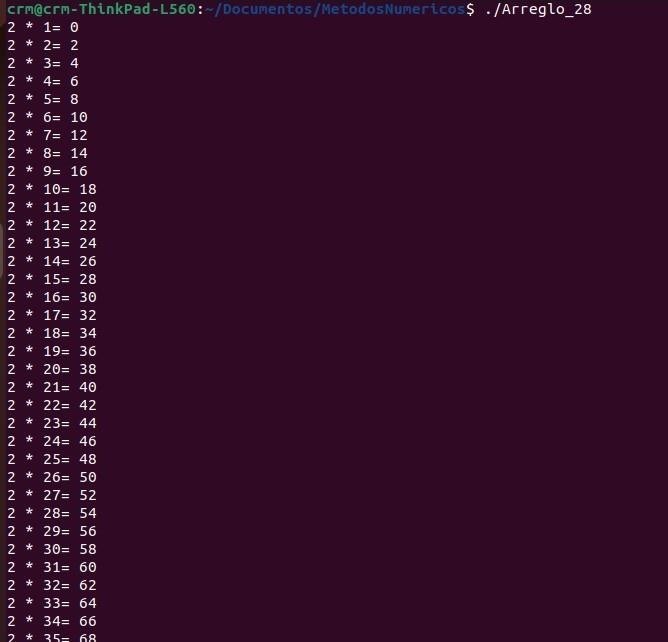
# Ejercicio 24 “Arreglos”

Imprimir el arreglo de 10 elementos sin ayuda de ciclos.



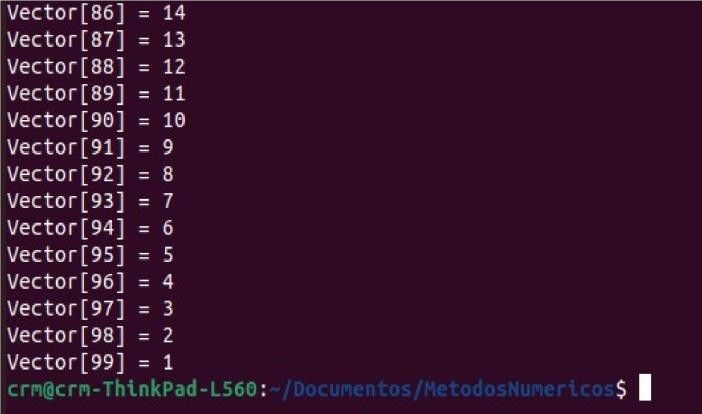
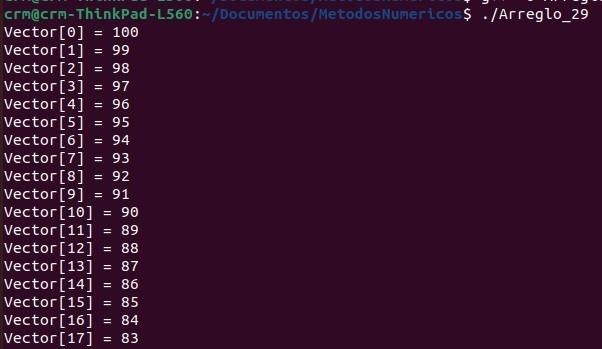
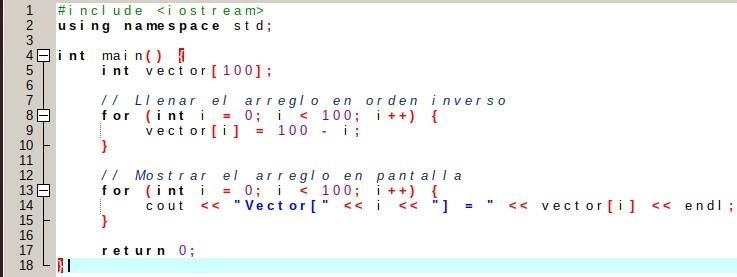
# Ejercicio 25 “100 Posiciones”

Crear un arreglo de 100 posiciones y llenar el arreglo con la tabla del dos y mostrar el arreglo en pantalla utilizando la estructura de los arreglos.



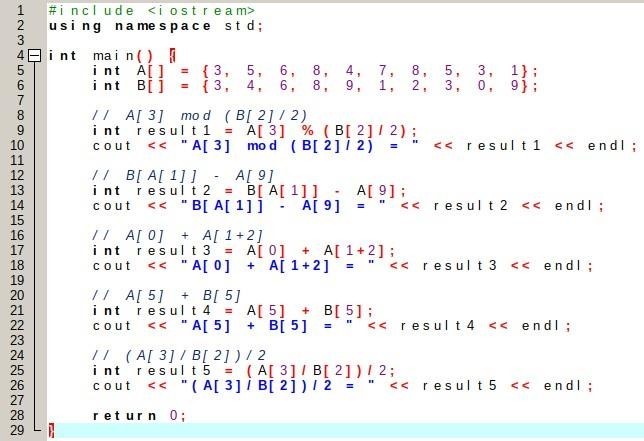
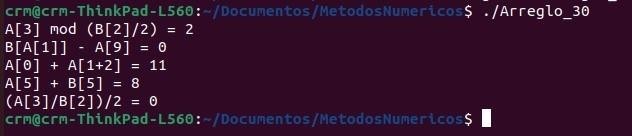
# Ejercicio 26 “Arreglo Inverso”

Crear un arreglo de 100 posiciones y llenar el arreglo al orden inverso al indice y mostrar el arreglo en pantalla usando la jerarquía de los arreglos.



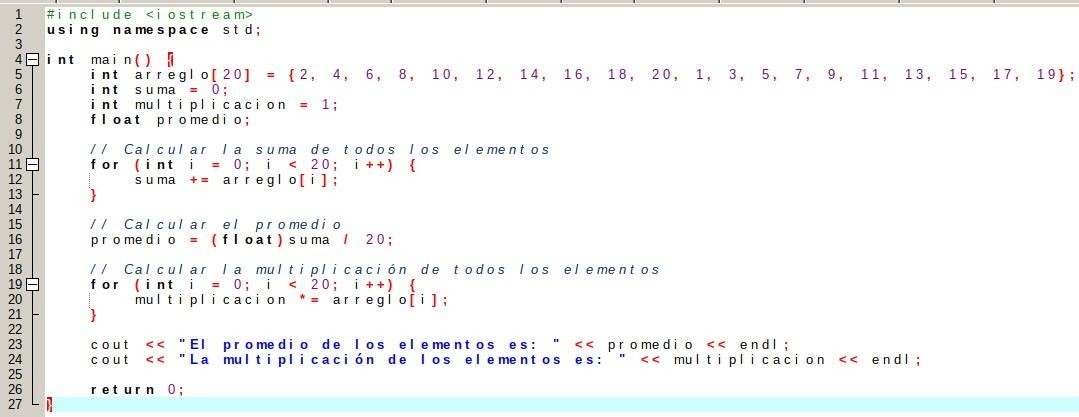
# Ejercicio 27 “Operaciones Aritméticas”

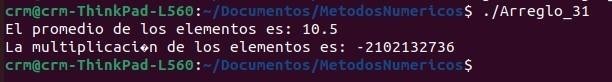
Dados un par de arreglos realizar 5 operaciones aritméticas y obtener los resultados planteados.



# Ejercicio 28 “Arreglo de 20 Posiciones”

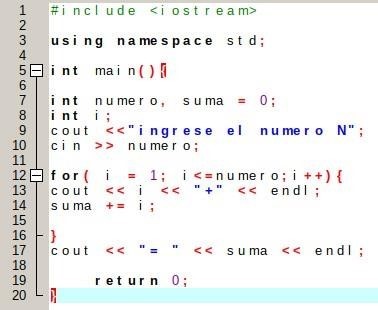
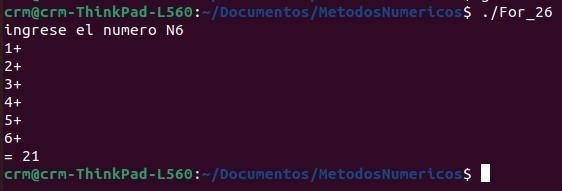
Crea un arreglo 20 posiciones. Asígnale a cada elemento un valor. Calcula el promedio de todos los elementos y calcula la multiplicación de todos los elementos.





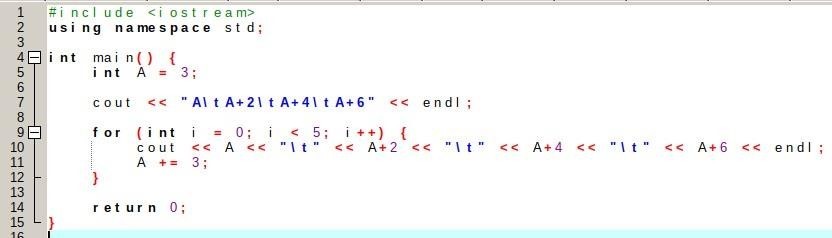
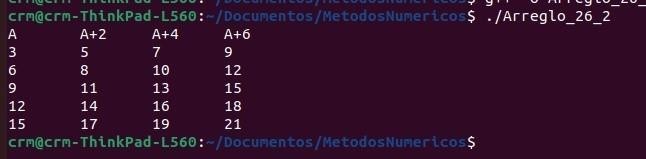
# Ejercicio 29 “Arreglo Suma”

Escriba un programa que reciba un numero N del usuario y haga la suma de todos los números desde 1 hasta N, usando la estructura del ciclo FOR.



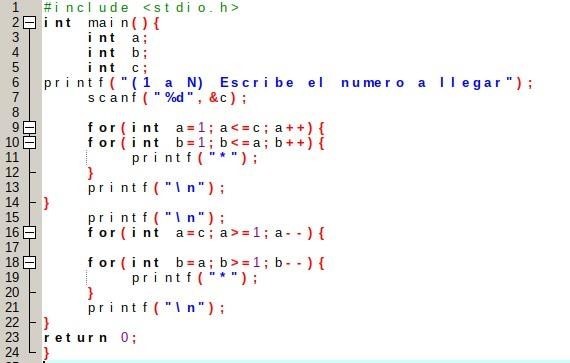
# Ejercicio 30 “Tabla de Valores”

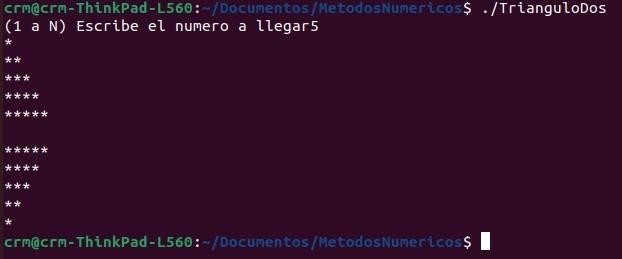
Escriba un programa en C que utilice un ciclo para reproducir una tabla de valores especifica como la del ejemplo planteado.



# Ejercicio 31 “Escalinata de asteriscos”

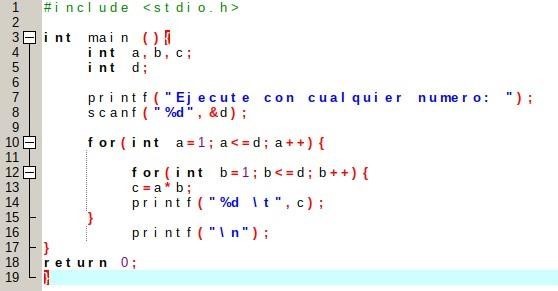
Escribir un programa que reciba un número N del usuario e imprima una escalinata de N pisos de asteriscos ascendente y descendente.

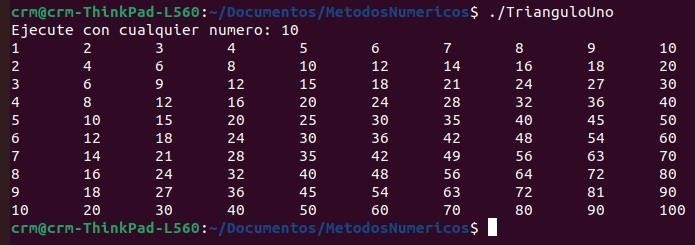




# Ejercicio 32 “Tabla de Multiplicar”

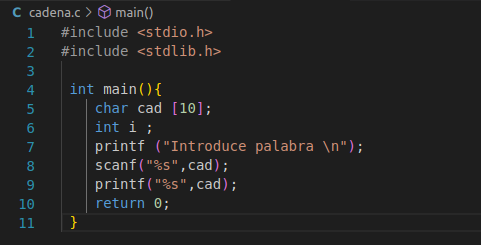
Realizar un programa que muestre las tablas de multiplicar al ingresar un valor desde el teclado mediante el ciclo “FOR”.

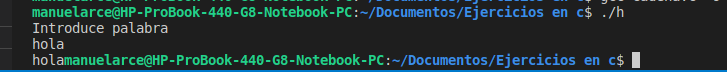




# Ejercicio 33 “Declarar una cadena”

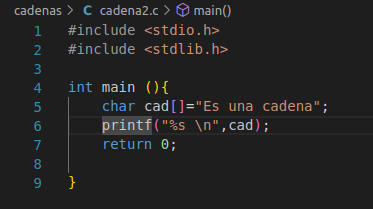
**Realizar un programa que muestre un arreglo de caracteres donde por lo general el ultimo elemento se a 0**

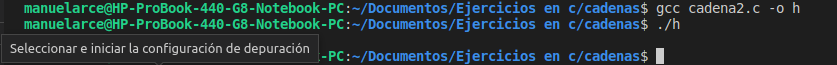
****

****

# Ejercicio 34 “Declarar una cadena”

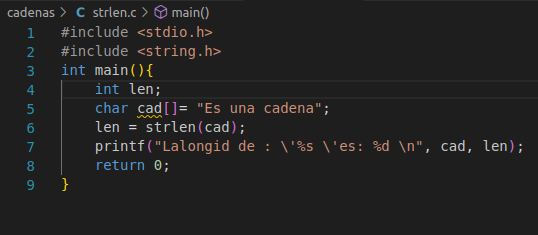
**Realizar un programa que muestre un arreglo de caracteres donde por lo general el ultimo elemento se a 0**

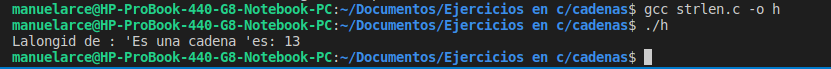
****

****

# Ejercicio 35 “Ejercicio Strlen”

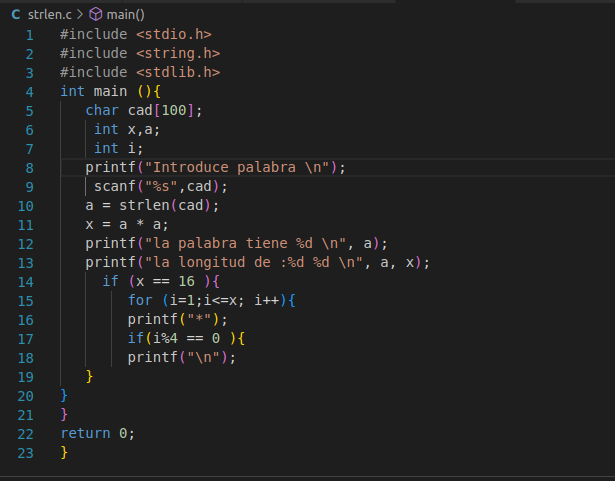
**Devuelve la longitud de una cadena sin tomar en cuenta sin tomar en cuneta el valor final**

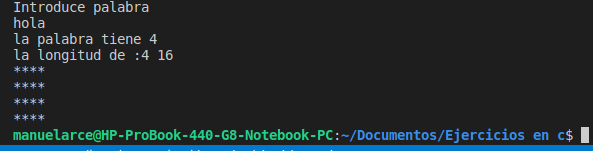
****

****

# Ejercicio 36 “Ejercicio Strlen”

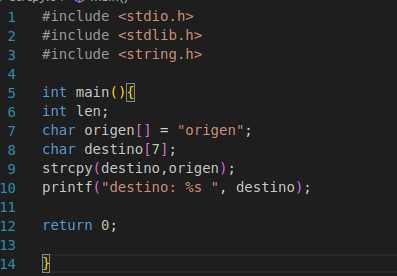
**Devuelve la longitud de una cadena sin tomar en cuenta sin tomar en cuneta el valor final e imprimir esa cantidad de caracteres en una tabla con asteriscos**

****

****

# Ejercicio 37 “Ejercicio Strcpy”

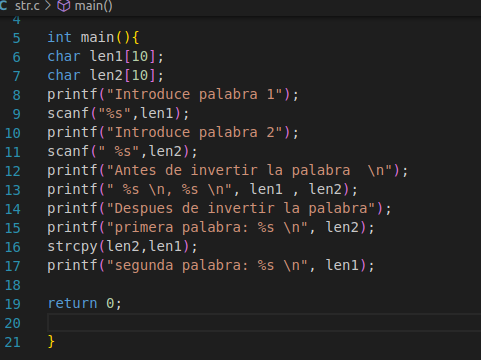
**Lee una cadena y la cambia de cadena origen a cadena destino**

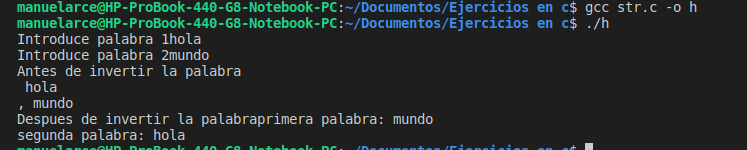
****

****

# Ejercicio 38 “Ejercicio Strcpy”

**Lee una cadena y la cambia de cadena origen a cadena destino**

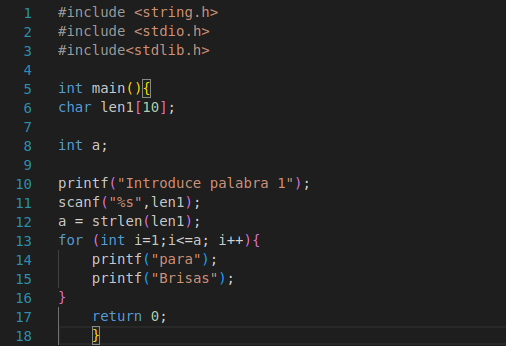
****

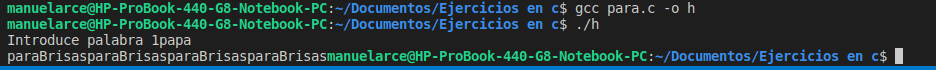
****

# Ejercicio 37 “Ejercicio Strcpy”

**Lee una cadena y la cambia de cadena origen a cadena destino utilizando la palabra**

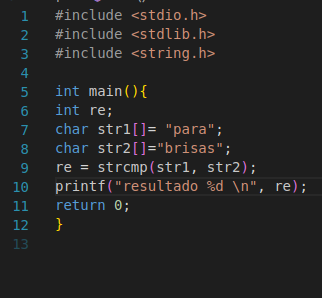
**Para y brisas**

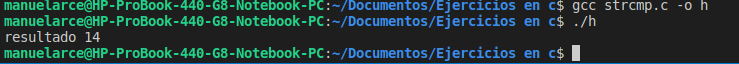
****

****

# Ejercicio 38 “Ejercicio Strcmp”

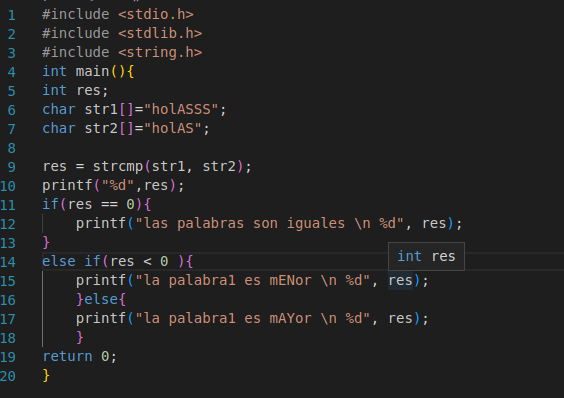
**Lee una cadena y la compara devolviendo un 0 si son iguales**

****

****

# Ejercicio 38 “Ejercicio Strcmp”

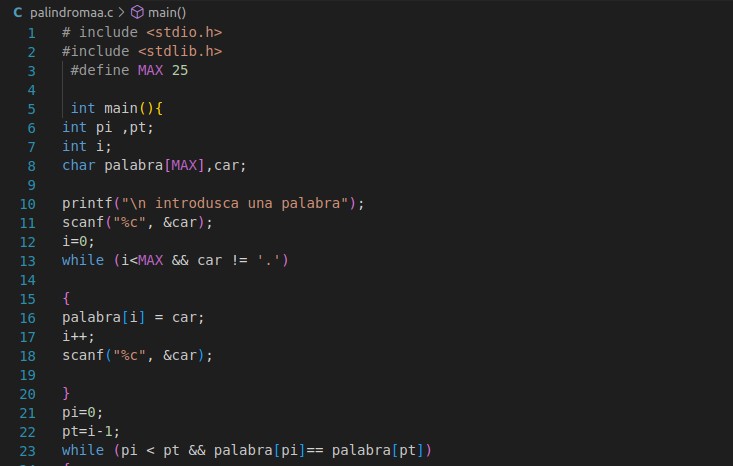
**Lee una cadena y la compara evaluando cual es mas grande**

****

****

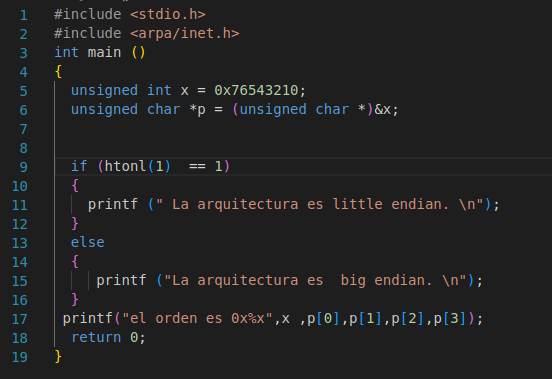
# Ejercicio 39 “Ejercicio Strcmp”

**Realiza un ejercicio que determine si una palabra es palindroma**

****

# Ejercicio 40 “Ejercicio Arquitectura”

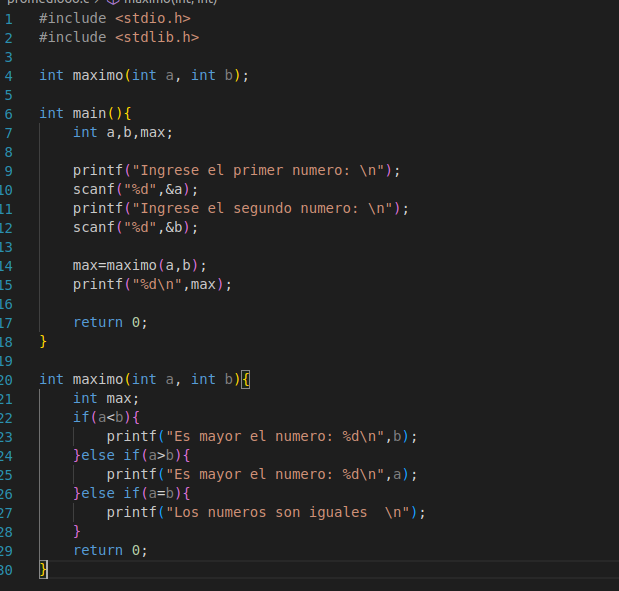
**Realiza un ejercicio que determine la arquitectura**

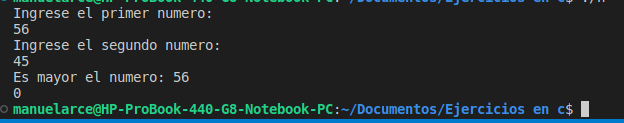
****

****

# Ejercicio 41 “Ejercicio Funciones ”

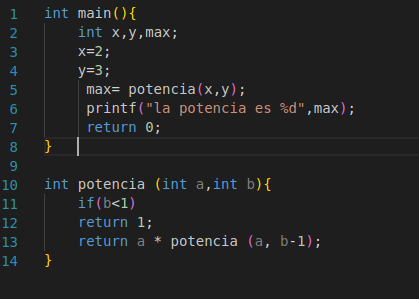
**Realiza un ejercicio que determine cual es mayor**

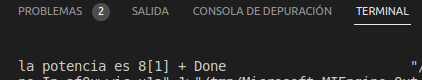
****

****

# Ejercicio 42 “Ejercicio Funciones”

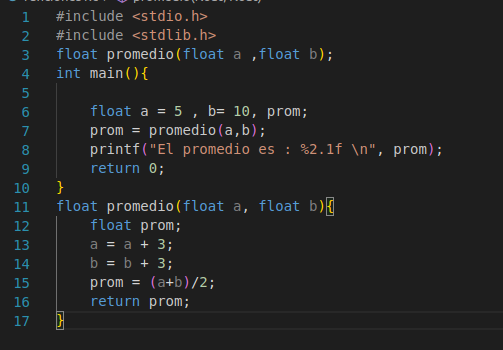
**Realiza un ejercicio que determine una función ejercicio 1**

****

****

# Ejercicio 42 “Ejercicio Funciones”

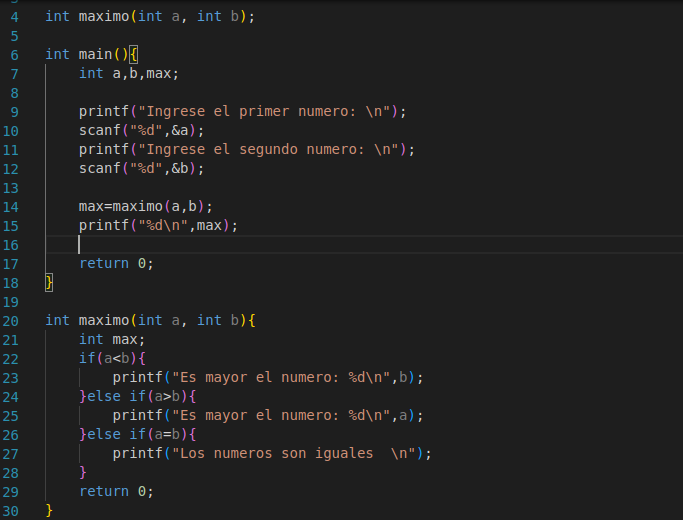
Realiza un ejercicio que determine una función ejercicio 2

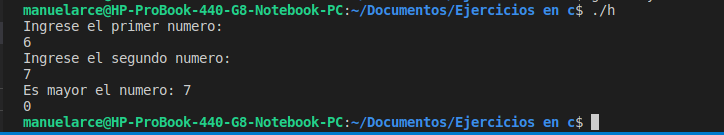
****

****

# Ejercicio 43 “Ejercicio Funciones”

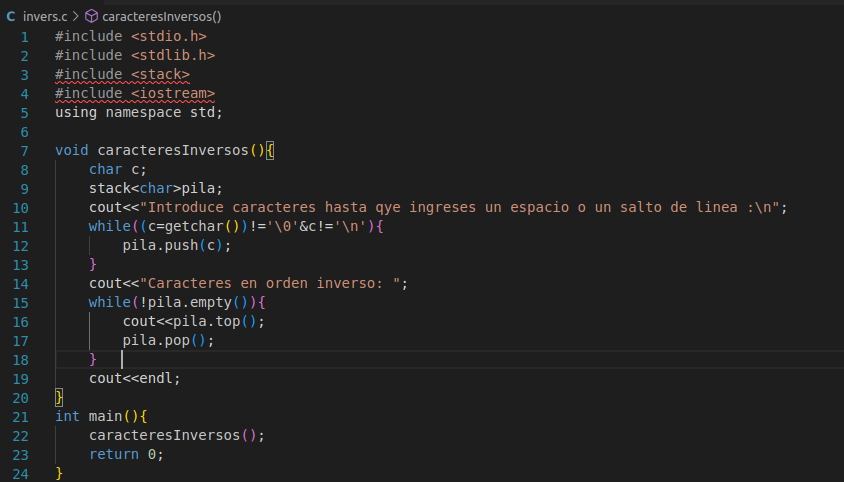
Realiza un ejercicio que determine una función ejercicio 3 regresa el mayor

****

****

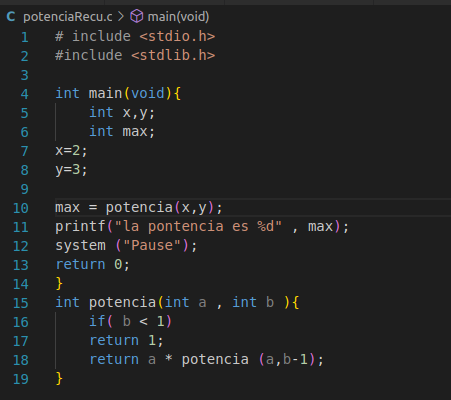
# Ejercicio 44 “Ejercicio Funciones”

Realiza un ejercicio que determine una función ejercicio 4 invertir palabra

****

# Ejercicio 45 “Ejercicio Funciones”

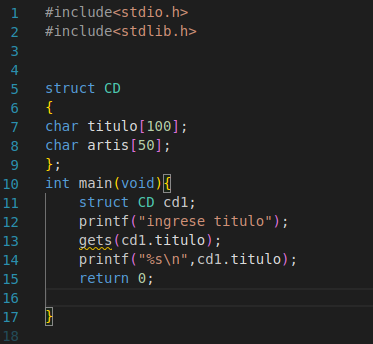
Realiza un ejercicio que determine una función recursiva



****

# Ejercicio 45 “Ejercicio Estructuras”

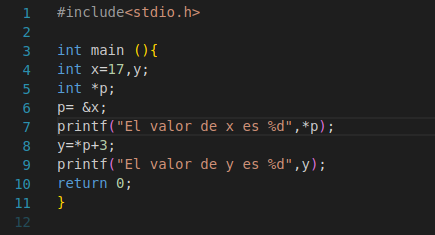
Realiza un ejercicio que represente el uso de estructuras



****

# Ejercicio 46 “Ejercicio Apuntadores”

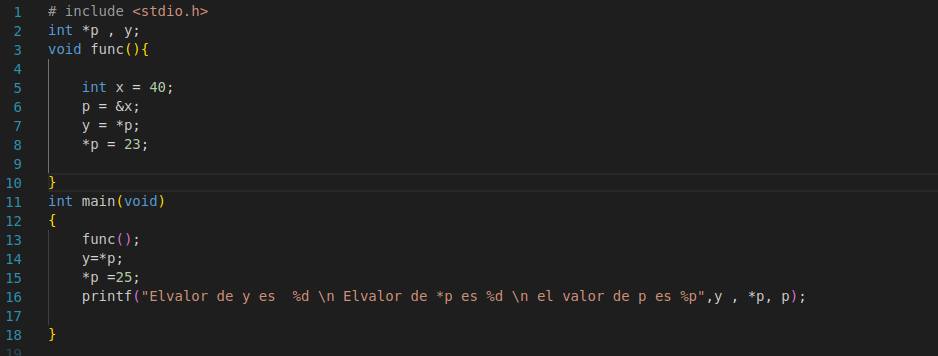
Realiza un ejercicio que represente el uso de apuntadores



****

# Ejercicio 47 “Ejercicio Apuntadores”

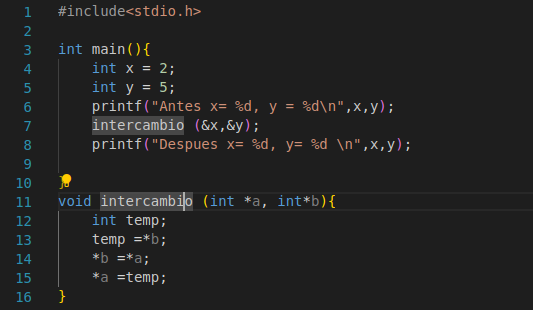
Realiza un ejercicio que represente el uso de apuntadores valor de \*p

****

****

# Ejercicio 48 “Ejercicio Apuntadores”

Realiza un ejercicio que represente el uso de apuntadores con cambio

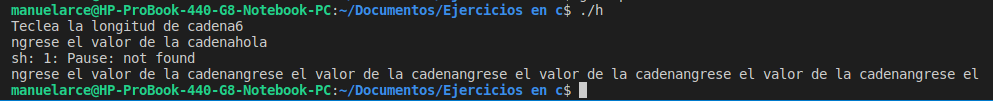


****

# Ejercicio 49 “Ejercicio Apuntadores”

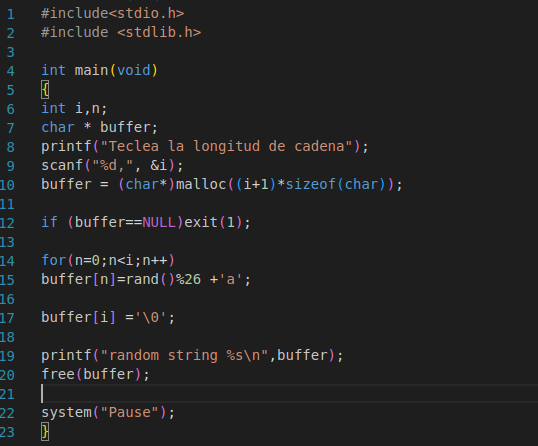
Realiza un ejercicio que represente el uso de malloc

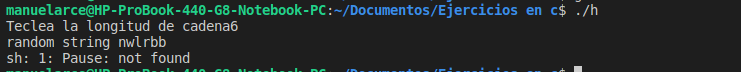
****

****

# Ejercicio 50 “Ejercicio Apuntadores”

Realiza un ejercicio que represente el uso de malloc y sizeof

****



**V. Conclusiones**

En esta unidad abarcamos temas muy interesantes, desde la manera elaborar un algoritmo en lo que fue un nuevo lenguaje de programación, hasta el aprender a manipular los distintos ciclos de programación que conocemos y del mismo modo abordamos el tema de arreglos, todos y cada uno de estos elementos de programación aunados a lo que es el manejo de números, en su mayoría los ejercicios son simples pero requieren de una buena capacidad de razonamiento, análisis y sobre todo de la capacidad para llevar los conocimientos de la teoría a la práctica, considero que con estos ejercicios me di cuenta de algunas habilidades de que necesito reforzar y así mismo temas que necesito repasar para poder comprender los en sus totalidad para llevarlos a la práctica.