|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Nombre de la práctica** | **Ejercicios lenguaje c** | | | **No.** | **1** |
| **Asignatura:** | **Métodos Númericos** | **Carrera:** | **INGENIERIA EN SISTEMAS COMPUTACIONALES** | **Duración de la práctica (Hrs)** |  |

**I. Competencia(s) específica(s):**

**II. Lugar de realización de la práctica (laboratorio, taller, aula u otro):**

Aula

**III. Material empleado:**

Compilador de c (gcc)

Visual studio code (Plataforma Linux)

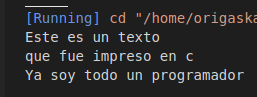
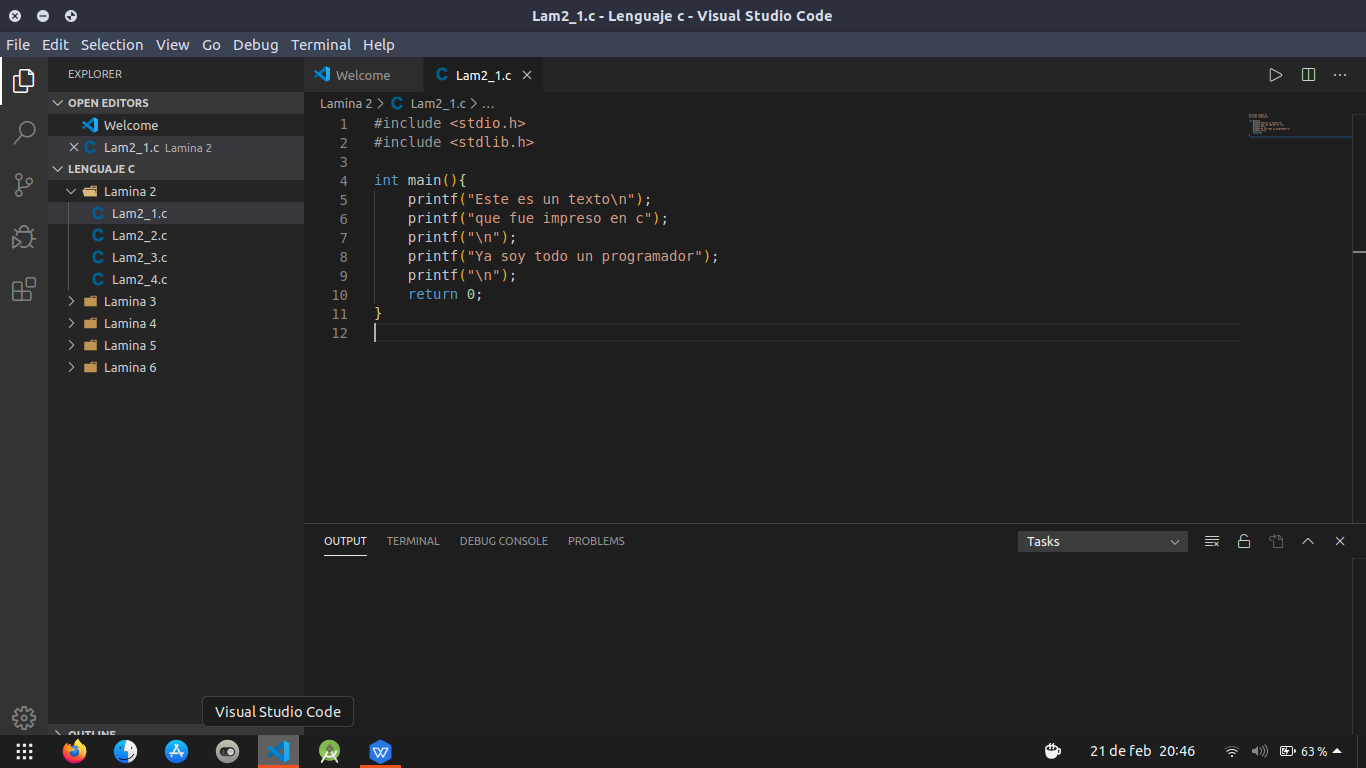
**IV. Desarrollo de la práctica:**

Comenzamos los ejercicios aprendiendo a mostrar mensajes en pantalla con la funcion **printf**

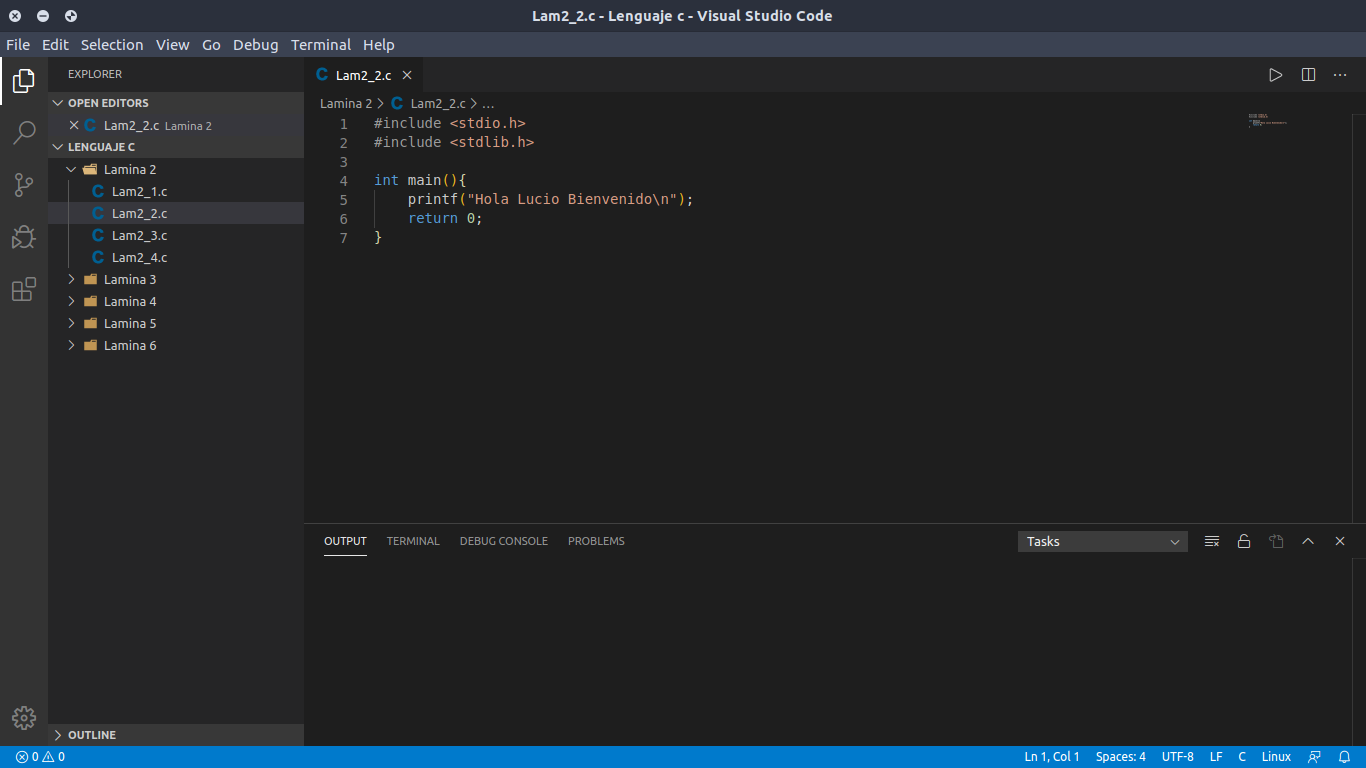
para poder usar esta funcion debemos incluir en el proyecto la Biblioteca **stdio.h**.

Usamos el caracter especial **\n** para realizar un salto de linea.

El codigo es escrito en la clase principal que en este caso esta retornano 0 al finalizar su ejecución.

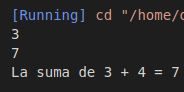
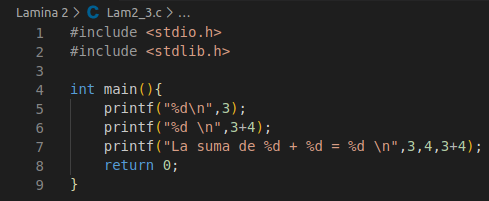


En el segundo programa realizamos algo muy similar al primero, en este mostramos un saludo con nuestro nombre en la pantalla.

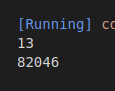
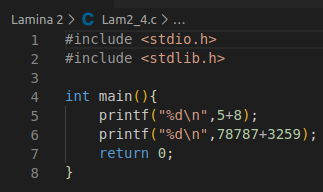
Captura de pantalla de 2020-02-21 22-40-57

Para el tercer programa usamos nuevos complementos para el codigo, uno de ellos es un especificador de acceso **%d** para poder mostrar dentro de printf un numero decimal.

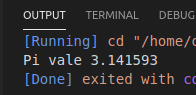
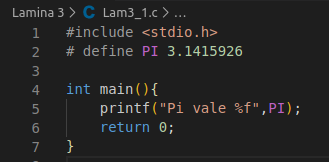
Otro termino es el simbolo de adición **+** con el cual realizamos sumas de valores numericos.



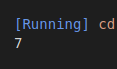
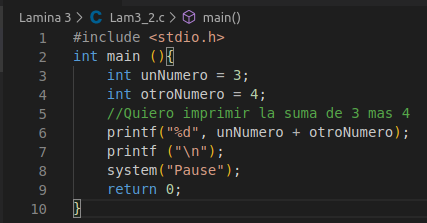
En el siguiente programa realizamos dos sumas simples de numeros enteros y podemos notar que **%d** funciona para cualquier tipo de dato Entero.



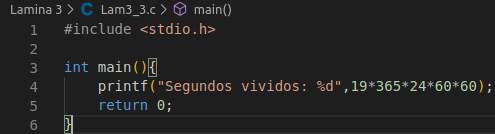
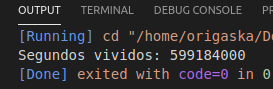
En el siguiente ejericio realizamos la declaracion de una constante usando la palabra #**define** y mostrando que se puede realizar una impresion de numeros de punto flotante con **%f**.

****

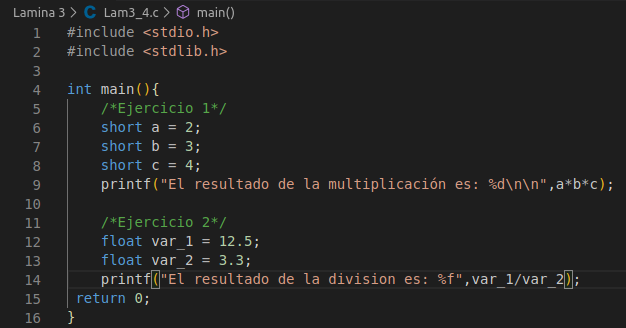
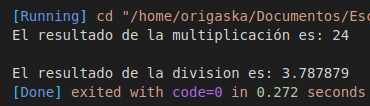
Para el siguiente ejercicio realizamos la declaracion de dos variables de tipo entero con valores distintos a 0 y mostramos la suma de estos dos números.



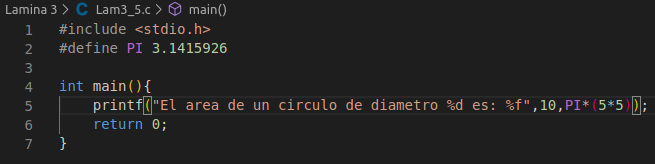
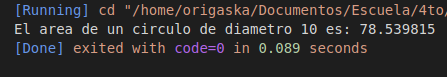
Realizamos un progra,a que determine los segundos vividos segun nuestra edad, para ello ocupamos la formula años\*365\*24\*60\*60



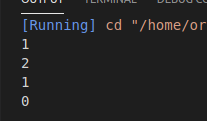
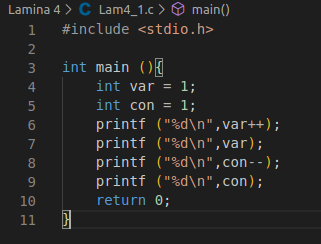
Para el siguiente ejercicio realizamos la suma de 3 variables de tipo entero y la divicion de dos numeros flotantes que tambien mostramos en pantalla.



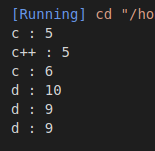
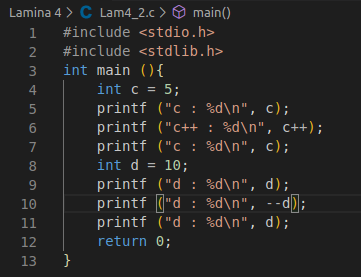
En el siguiente programa mostamos el resultado del area de un circulo de diametro de 10 unidades.



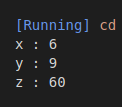
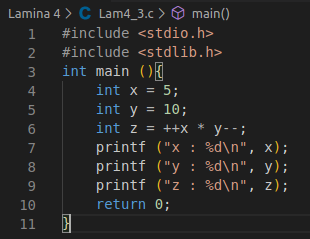
En el siguiente programa realizamos post incremento, tambien realizamos post decremento para poder visualizar en pantalla para poder entender mejor estos tipos de operaciones que aumentan los valores despues de ser utilizada.



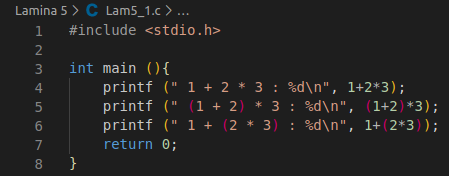
Para el siguiente programa Analizamos el uso de las operaciones pre decremento y pre incremento ya que esta operacion ocurre antes de ocupar la variable.

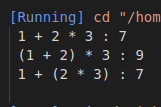


En el ejemplo 3 de la lamina realizamos operaciones con los conceptos vistos en los dos programas anteriores con las variables x y y para luego mostrarlas junto con la variable z donde se aplicaron las operaciones de las otras 2 variables.

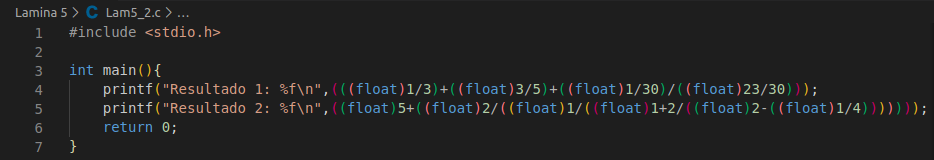


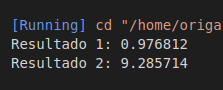
En la soquiente lamina vemos la precedencia de operadores para comenzar a entender estas operaciones realizamos operaciones que mostramos en pantalla con distinto orden de un grupo de parentesis que es el operador de mayor jerarquia.



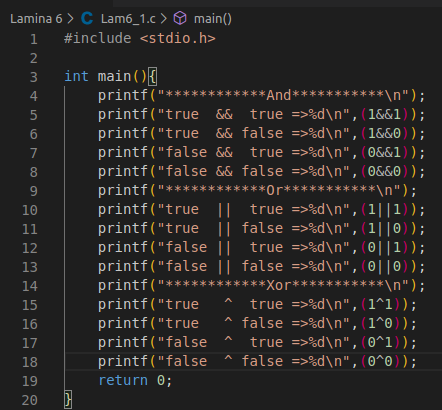


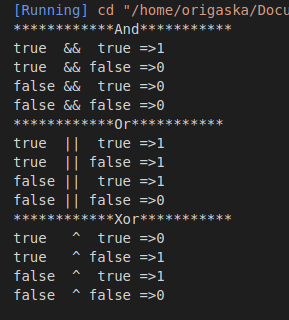
En el siguiente ejercicio realizamos dos diviciones, en este caso al utilizar numeros enteros en una divicion nos entrega un numero entero como resultado, para obtener un dato mas exacto necesitamos realizar un casting a los datos o declararlos directamente como flotantes, en este caso uso la opcion de casting a tipo float.



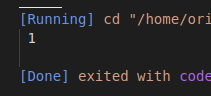
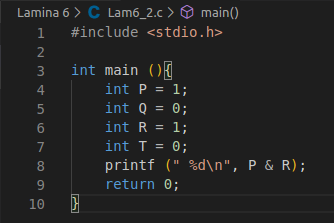


El siguiente ejercicio realizamos las tablas de verdad para AND, OR y XOR.





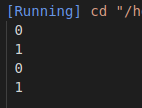
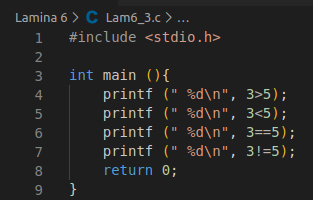
En este programa realizamos una comparacion logica con AND (&) y lo mostramos en pantalla.



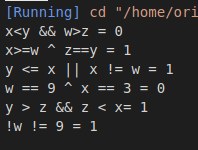
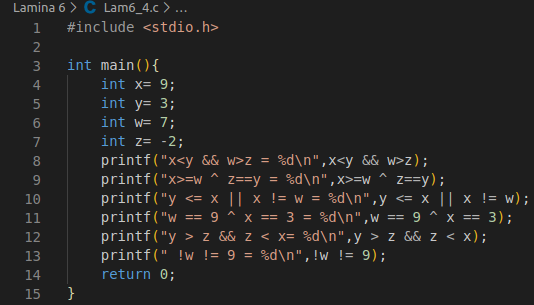
Las expresiones true y false se representan con 1 y 0 respectivamente.

Tambien podemos realizar operaciones con comparaciones logicas, esto seria con los operadores >,<,==,etc.

El siguiente programa mustra en pantalla el resultado de algunas operaciones con estos operadores.



Como ultimo programa de la lamina 6 realizamos un programa donde ocupamos los operadores logicos y las comparaciones logicas.



**V. Conclusiones:**

Aprender las funciones basicas de un lenguaje de programación de una forma practica nos ayuda a comprender mejor el lenguaje, programar en lenguaje c nos ayuda a poder realizar programas para poder controlar dispositivos ya sea como lenguaje de alto nivel o de vajo nivel, ademas que la forma de trabajarlo nos ayuda a desarrollar nuestras habilidades para codificar de una forma mas rapida.