|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Nombre de la práctica** | **Ejercicios lenguaje c** | | | **No.** | **1** |
| **Asignatura:** | **Métodos Numéricos** | **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)

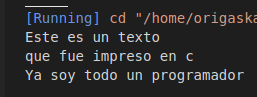
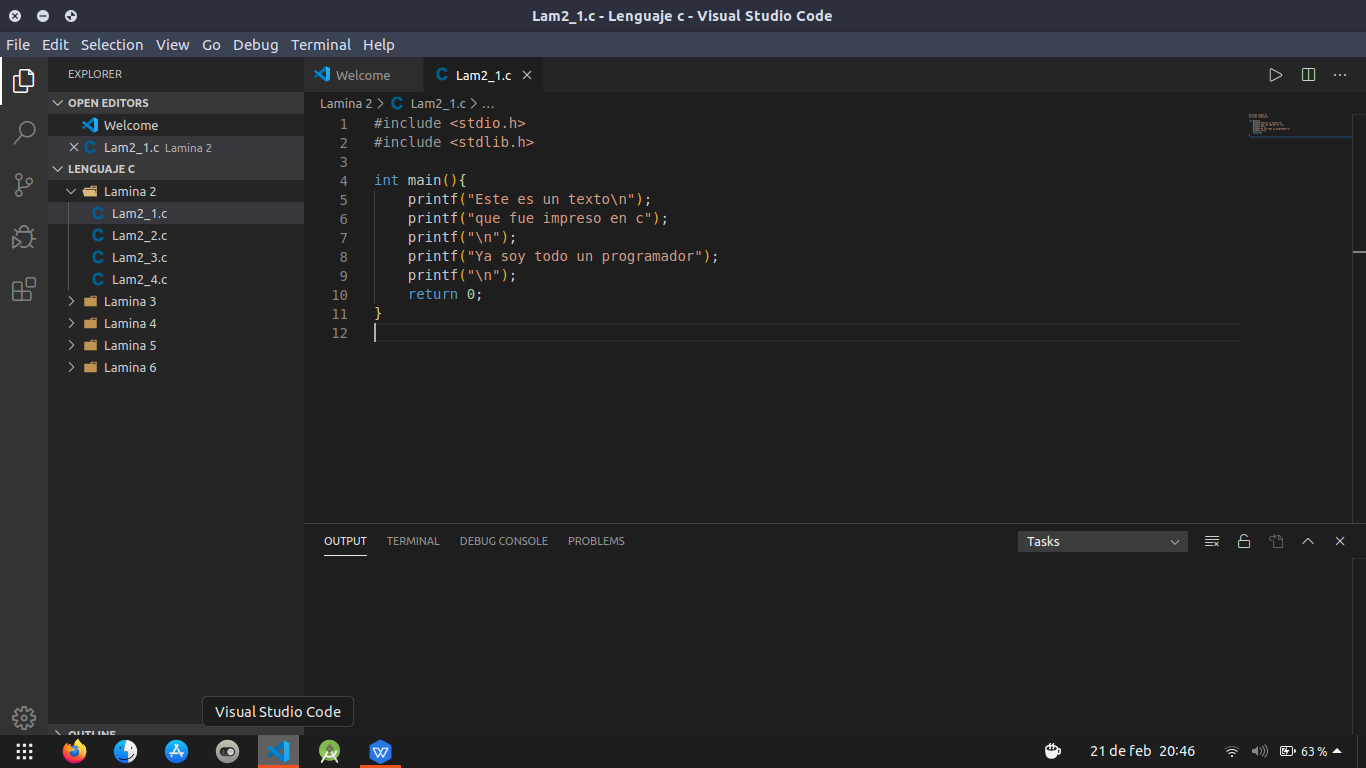
**IV. Desarrollo de la práctica:**

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

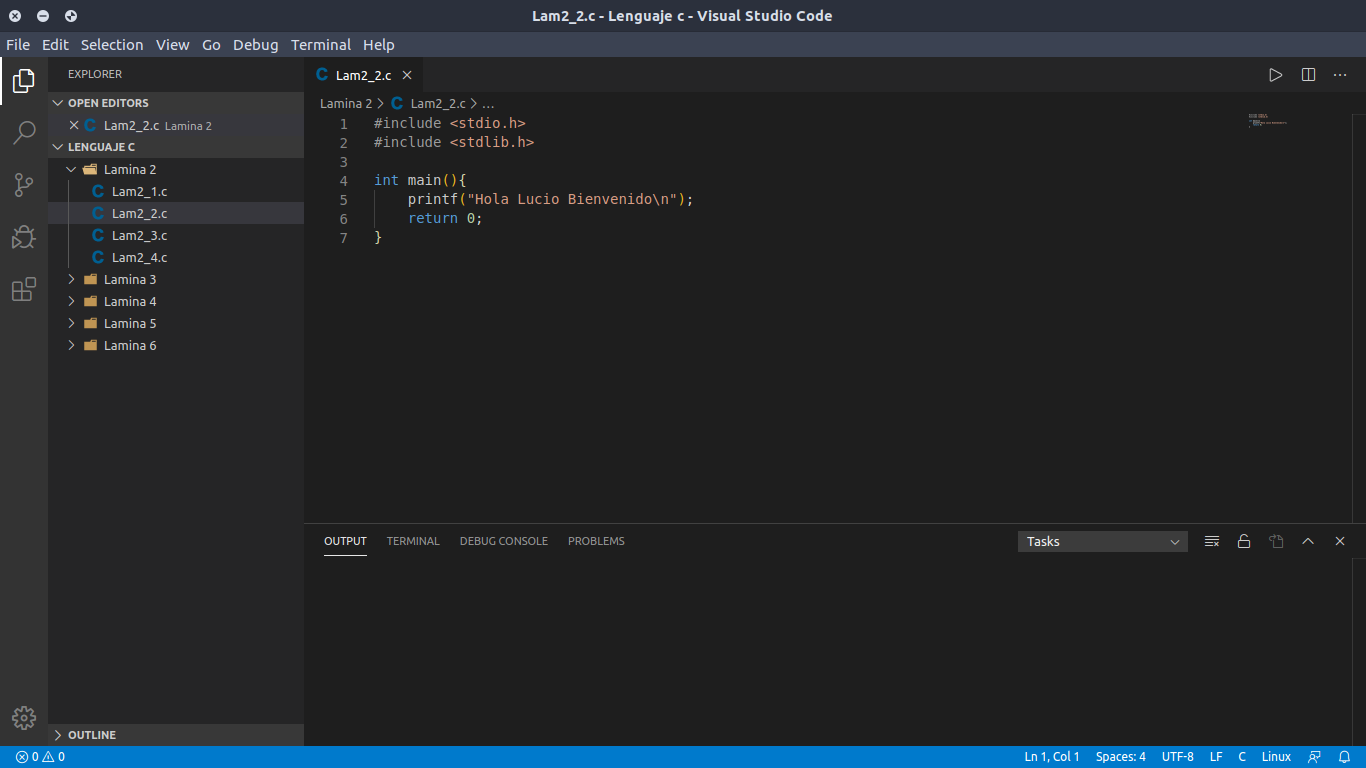
para poder usar esta función debemos incluir en el proyecto la Biblioteca **stdio.h**.

Usamos el carácter especial **\n** para realizar un salto de línea.

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

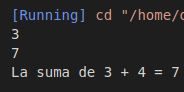
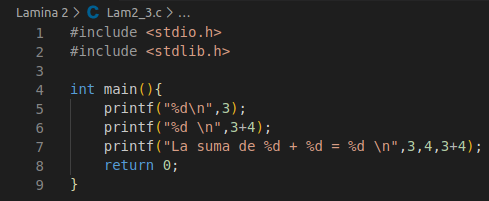


En el segundo programa realizamos algo muy similar al primero, en este mostramos un saludo con nuestro nombre en la pantalla.

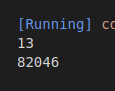
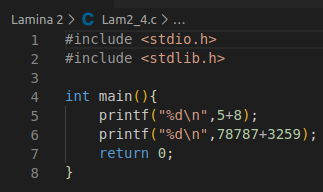
Captura de pantalla de 2020-02-21 22-40-57

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

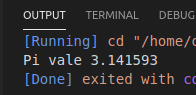
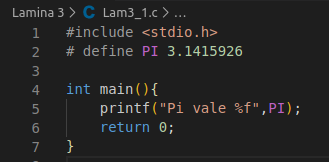
Otro termino es el símbolo de adición **+** con el cual realizamos sumas de valores numéricos.



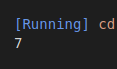
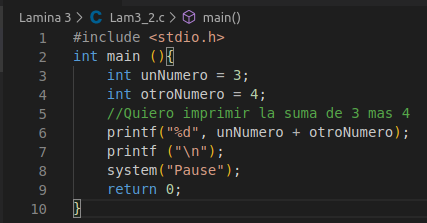
En el siguiente programa realizamos dos sumas simples de números enteros y podemos notar que **%d** funciona para cualquier tipo de dato Entero.



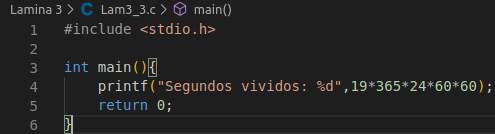
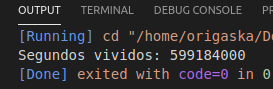
En el siguiente ejercicio realizamos la declaración de una constante usando la palabra #**define** y mostrando que se puede realizar una impresión de números de punto flotante con **%f**.

****

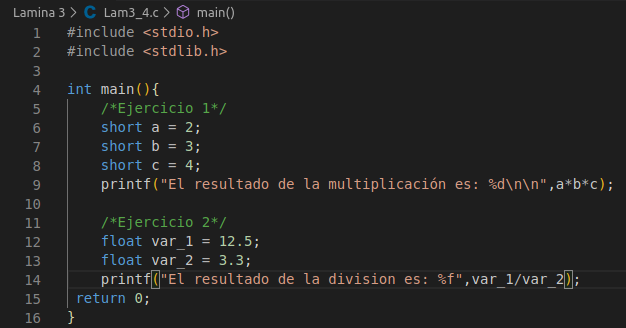
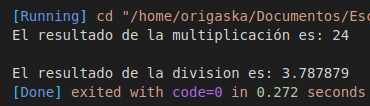
Para el siguiente ejercicio realizamos la declaración de dos variables de tipo entero con valores distintos a 0 y mostramos la suma de estos dos números.



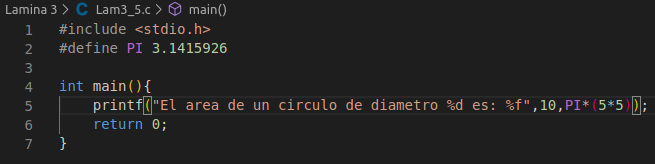
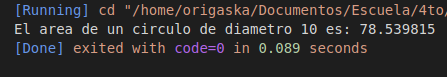
Realizamos un programa que determine los segundos vividos según nuestra edad, para ello ocupamos la formula años\*365\*24\*60\*60



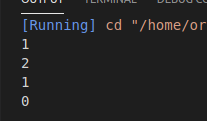
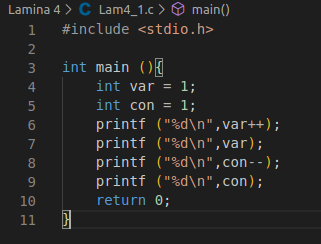
Para el siguiente ejercicio realizamos la suma de 3 variables de tipo entero y la división de dos números flotantes que también mostramos en pantalla.



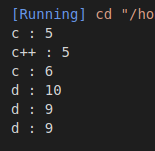
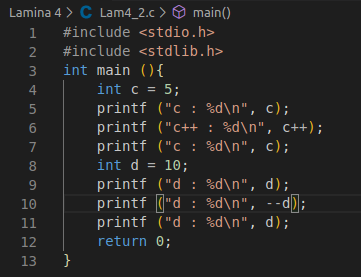
En el siguiente programa mostramos el resultado del área de un círculo de diámetro de 10 unidades.



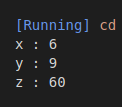
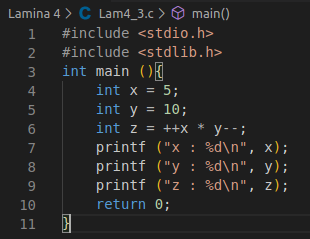
En el siguiente programa realizamos post incremento, también realizamos post decremento para poder visualizar en pantalla para poder entender mejor estos tipos de operaciones que aumentan los valores después de ser utilizada.



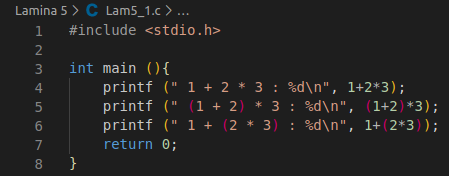
Para el siguiente programa Analizamos el uso de las operaciones pre decremento y pre incremento ya que esta operación ocurre antes de ocupar la variable.

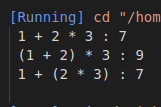


En el ejemplo 3 de la lámina realizamos operaciones con los conceptos vistos en los dos programas anteriores con las variables x y y para luego mostrarlas junto con la variable z donde se aplicaron las operaciones de las otras 2 variables.

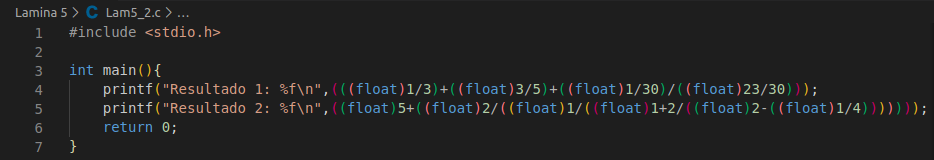


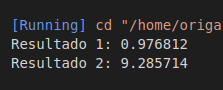
En la siguiente lamina vemos la precedencia de operadores para comenzar a entender estas operaciones realizamos operaciones que mostramos en pantalla con distinto orden de un grupo de paréntesis que es el operador de mayor jerarquía.



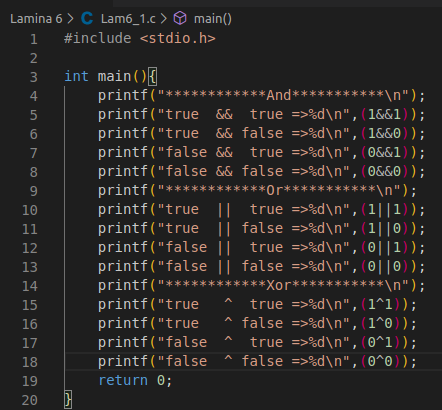


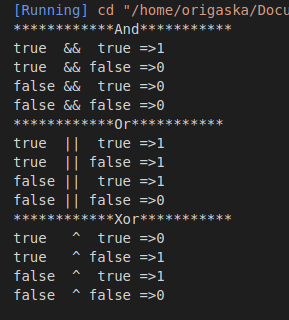
En el siguiente ejercicio realizamos dos divisiones, en este caso al utilizar números enteros en una división nos entrega un numero entero como resultado, para obtener un dato más exacto necesitamos realizar un casting a los datos o declararlos directamente como flotantes, en este caso uso la opción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



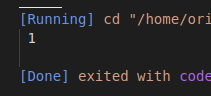
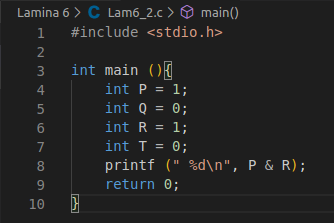


El siguiente ejercicio realizamos las tablas de verdad para AND, OR y XOR.





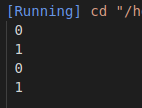
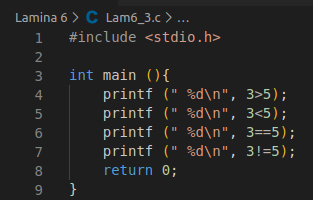
En este programa realizamos una comparación lógica con AND (&) y lo mostramos en pantalla.



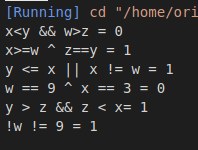
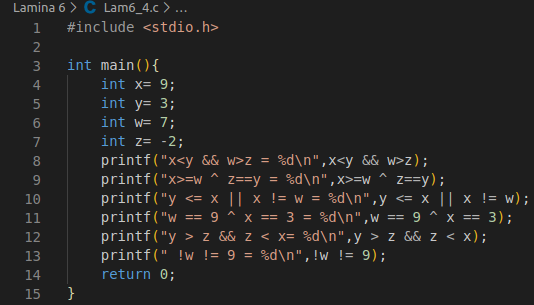
Las expresiones true y false se representan con 1 y 0 respectivamente.

También podemos realizar operaciones con comparaciones lógicas, esto sería con los operadores >, <, ==, etc.

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



Como último programa de la lámina 6 realizamos un programa donde ocupamos los operadores lógicos y las comparaciones lógicas.



**V. Conclusiones:**

Aprender las funciones básicas de un lenguaje de programación de una forma práctica 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 bajo nivel, además que la forma de trabajarlo nos ayuda a desarrollar nuestras habilidades para codificar de una forma más rápida.