|  |  |  |
| --- | --- | --- |
|  | **Carátula para entrega de prácticas** | |
| Facultad de Ingeniería | | Laboratorio de docencia |

Laboratorios de computación

salas A y B

|  |  |
| --- | --- |
| *Profesor:* | Adrian Ulises Mercado Martinez |
| *Asignatura:* | Fundamentos de programación |
| *Grupo:* | 15 |
| *No de Práctica(s):* | Práctica 7 |
| *Integrante(s):* | -Campos Lobatón Fernando  -Barrera Guardia Francisco Moises  -Ruiz Dávila Omar Jalil |
| *No. de Equipo de cómputo empleado:* | Equipos 5, 6 y 7 |
| *No. de Lista o Brigada:* | Brigada 1 |
| *Semestre:* | 2019-2020 |
| *Fecha de entrega:* | 03/10/2019 |
| *Observaciones:* |  |
|  |  |

CALIFICACIÓN: \_\_\_\_\_\_\_\_\_\_

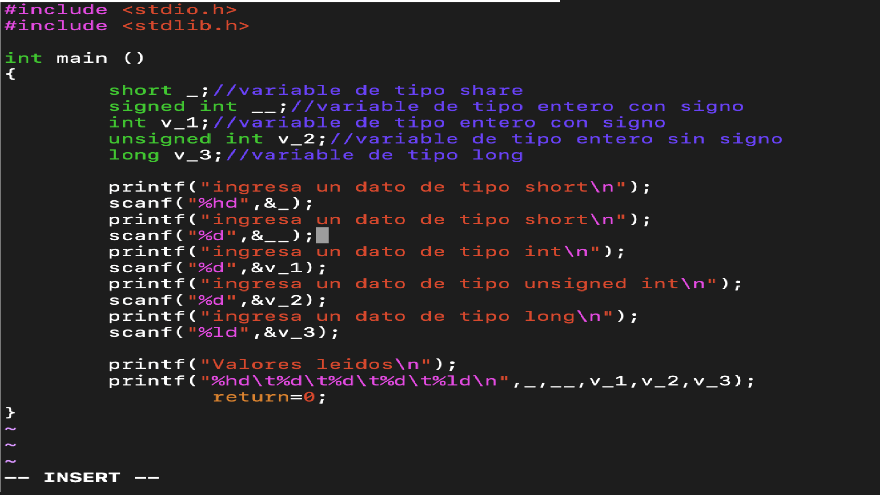
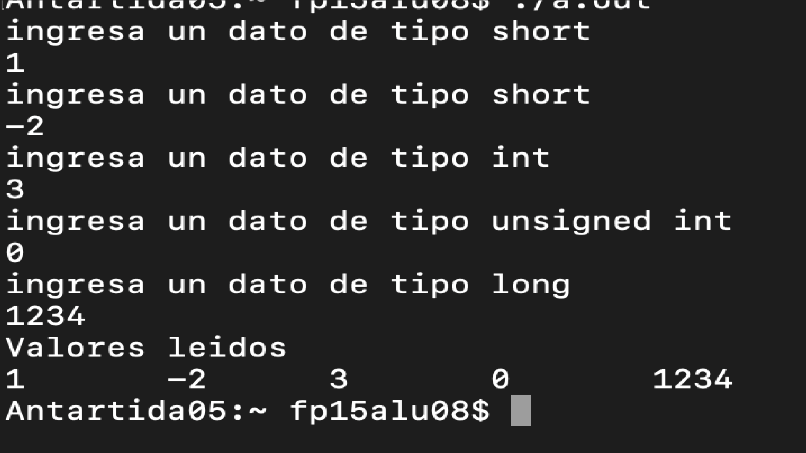
INTRODUCCIÓN

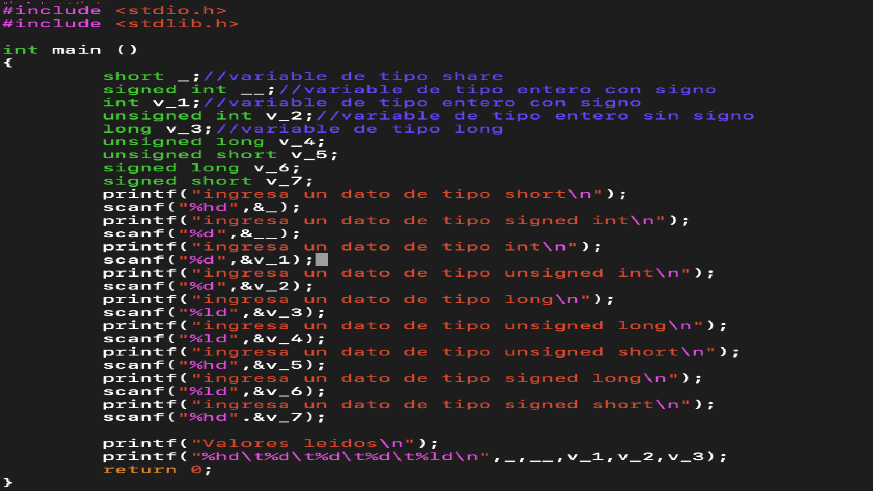
El lenguaje de programación C fue creado por Brian Kernighan y Dennis Ritchie a mediados de los años 70. La primera implementación del mismo la realizó Dennis Ritchie sobre un computador DEC PDP-11 con sistema operativo UNIX. C es el resultado de un proceso de desarrollo que comenzó con un lenguaje anterior, el BCPL, el cual influyó en el desarrollo por parte de Ken Thompson de un lenguaje llamado B, el cual es el antecedente directo del lenguaje C. El lenguaje C es un lenguaje para programadores en el sentido de que proporciona una gran flexibilidad de programación y una muy baja comprobación de incorrecciones, de forma que el lenguaje deja bajo la responsabilidad del programador acciones que otros lenguajes realizan por sí mismos. Así, por ejemplo, C no comprueba que el índice de referencia de un vector (llamado array en la literatura informática) no sobrepase el tamaño del mismo; que no se escriba en zonas de memoria que no pertenecen al área de datos del programa, etc. El lenguaje C es un lenguaje estructurado, en el mismo sentido que lo son otros lenguajes de programación tales como el lenguaje Pascal, el Ada o el Modula-2, pero no es estructurado por bloques, o sea, no es posible declarar subrutinas (pequeños trozos de programa) dentro de otras subrutinas, a diferencia de como sucede con otros lenguajes estructurados tales como el Pascal. Además, el lenguaje C no es rígido en la comprobación de tipos de datos, permitiendo fácilmente la conversión entre diferentes tipos de datos y la asignación entre tipos de datos diferentes.

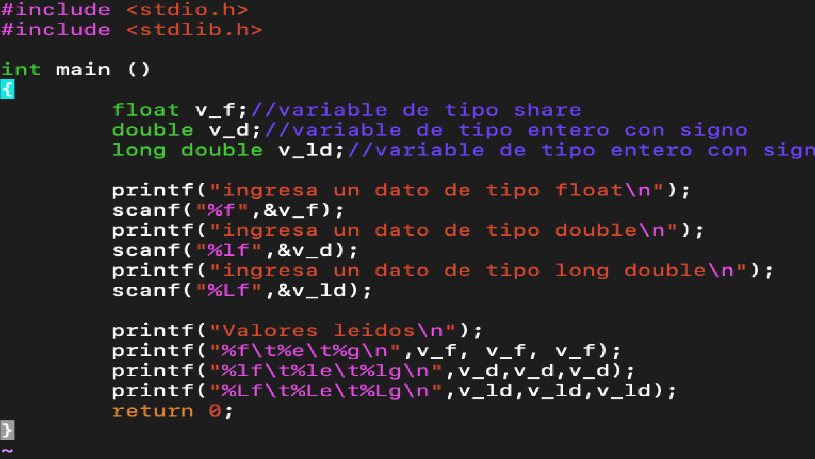
y en esta práctica programaremos con ese lenguaje y se harán resoluciones de problemas con los códigos correspondientes a la práctica, además de que se denotarán algunos códigos y estructuras fundamentales para poder hacer la resolución de problemas de una manera más efectiva.

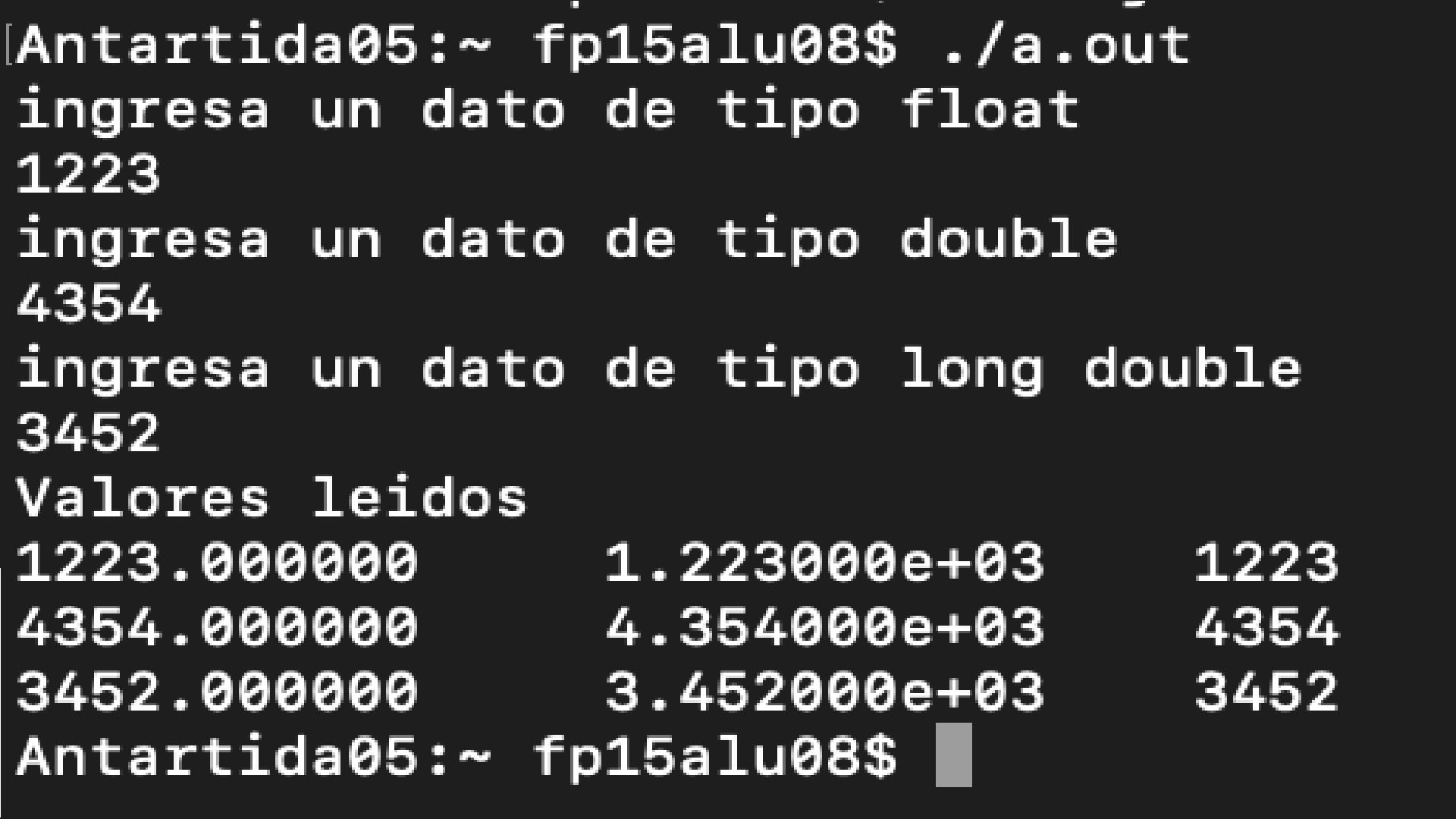
INTRODUCCIÓN

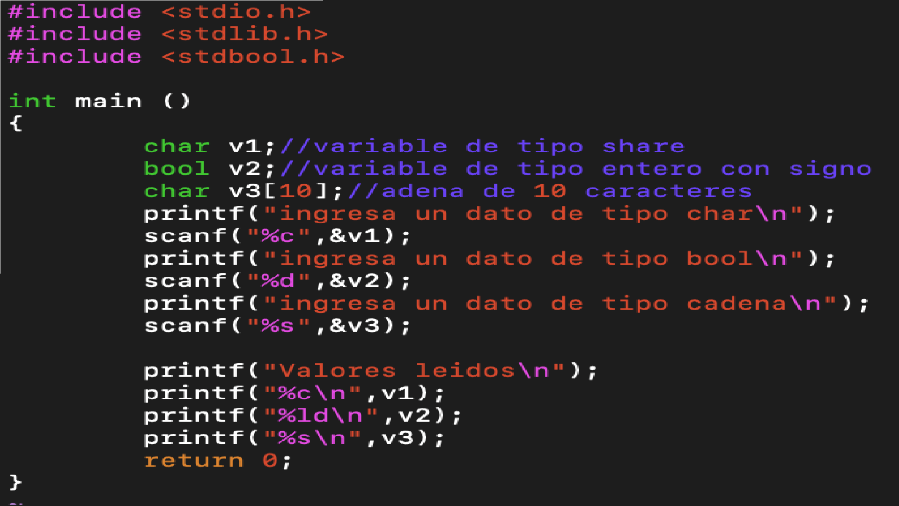
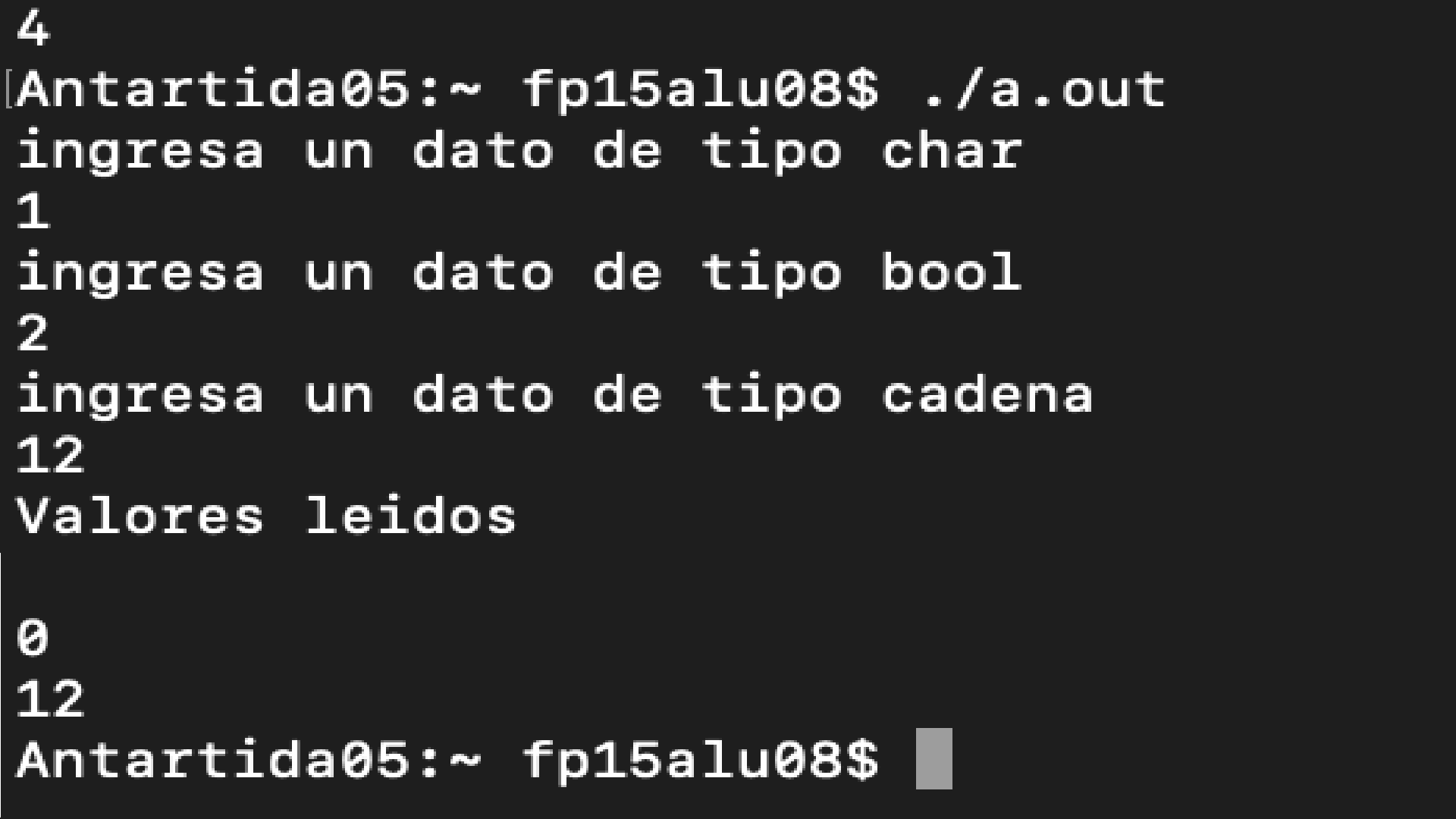
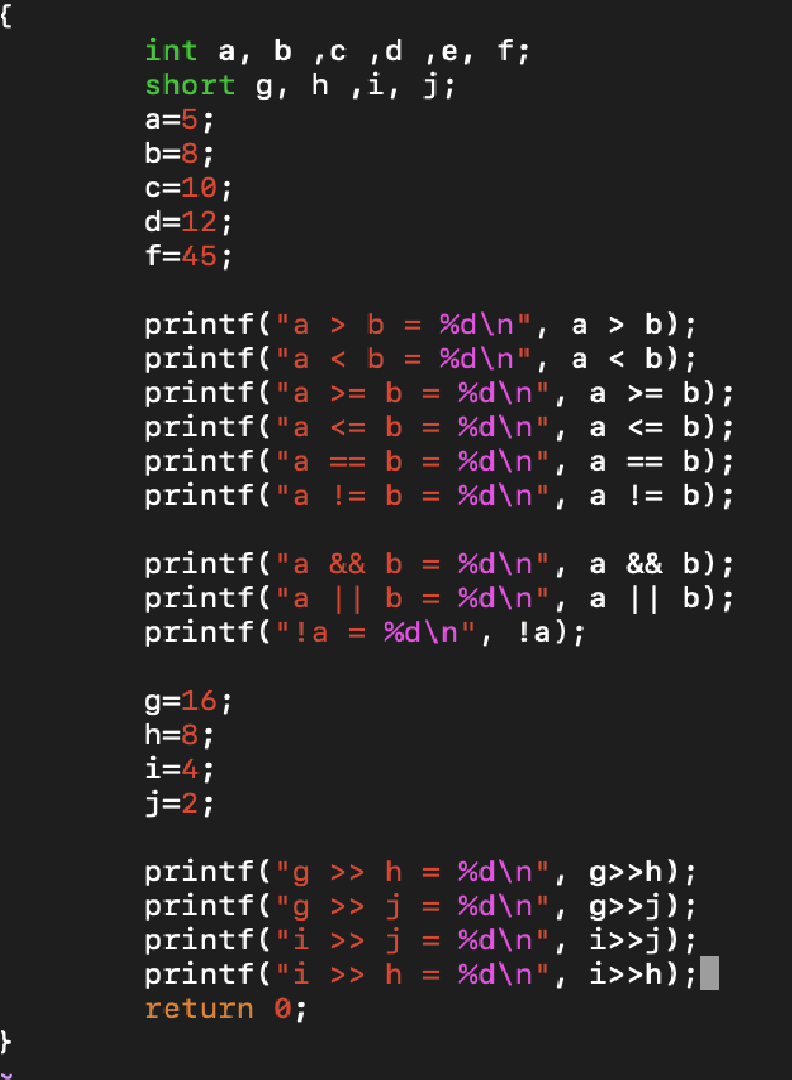
En esta práctica vimos diversos algoritmos para conocer mejor los distintos tipos de variables, así como sus valores máximos y mínimos, y lo signed y unsigned.

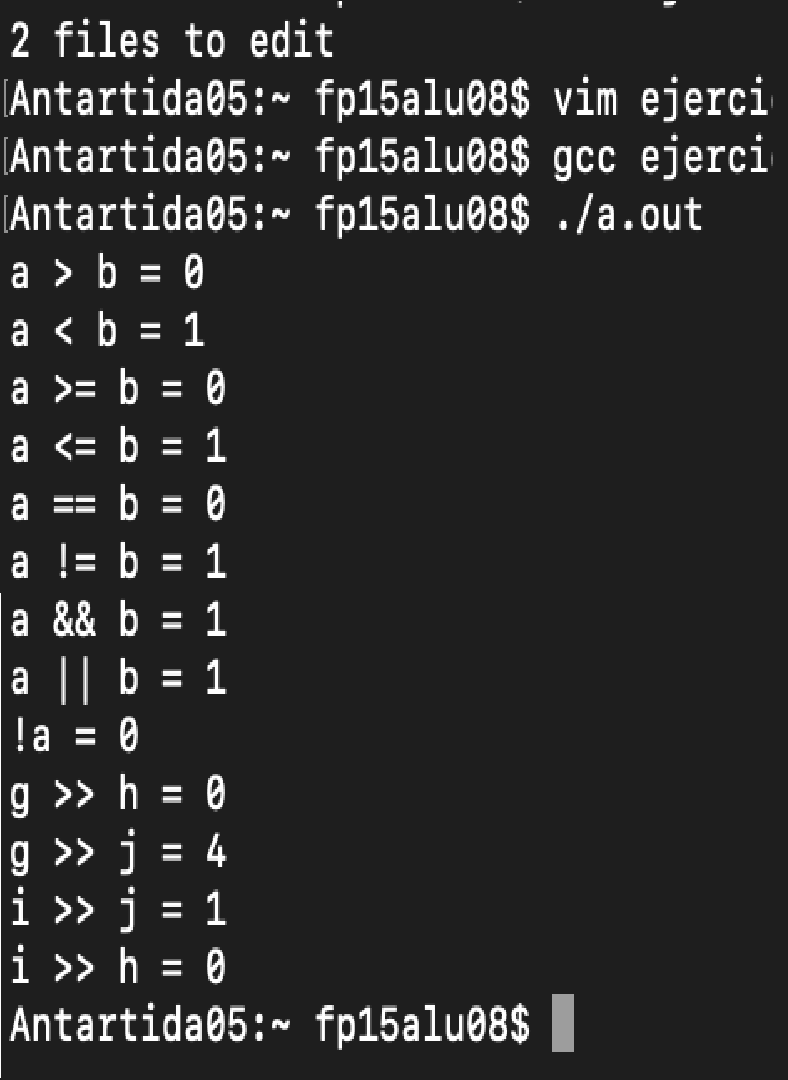
En este algoritmo podemos ver los tres tipos de variables más comunes que se utilizan al programar y los imprime al correr el programa. En la imagen de abajo se ve lo que pasa al correr el programa y como imprime los valores

En el de abajo se ve un programa similar, pero con más variables como long y short con unsigned y signed

En este otro, se imprimen valores usando las variables float, doublé y long doublé asi como el imprimirlos al dar un valor a cada uno.

Aquí se muestran los valores imprimidos de float, doublé y long doublé.

En este otro se muestran las variables de tipo bool y chara si como abajo muestra el cómo se imprimen estas variables, recordando poner un include stdbool hasta arriba ya que si no reconoce estas variables el programa.

Y por último hicimos un programa que utilizaba los operadores aritméticos para decirnos si un numero era mayor o igual, menor o igual, distinto de, igual a, y, o, y otros nuevos llamados operadores de corrimiento, así como imprimir los resultados de cada uno.

Al final puedo decir que cada una de las variables tiene su propio nivel de almacenamiento y en este va de –x a x, y también pudimos ver que, si se introduce un valor mayor al permitido por cada variable, al momento de imprimir sus valores te da el valor máximo de la variable con el signo contrario al del introducido por ejemplo si el número máximo y mínimo de una variable es -30, 30 pero introduces un 45, te imprimirá el -30.

CONCLUSIONES

Barrera Guardia Francisco Moises:

En esta práctica se pudo aprender el cómo se elabora un programa y el cómo se elabora de tal manera que los problemas fueran resueltos con eficacia utilizando los códigos más sencillos y fundamentales de la programación en tipo c, además de que se logró el objetivo utilizando varias maneras de códigos en los cuales se llegara a necesitar este tipo de lenguaje.

Campos Lobaton Fernando

Se Cumplió el objetivo de poder entender el valor de memoria de cada una de las siguientes variables que a la vez también son declaradas en un programa en forma de secuencia para mostrar valores de entrada y de salida.se entendió el funcionamiento de cada tipo de variable.

Ruiz Dávila Omar Jalil

En esta práctica pude empezar a reforzar mis conocimientos básicos al programar en C, además de comprender mejor las funcionalidades de cada variable y operador a la hora de realizar un programa más difícil de elaborar, así como el que sucede si introducimos un número mayor al permitido por la variable.