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

**LABORATORIOS DE COMPUTACIÓN**

|  |  |
| --- | --- |
| *Profesor:* | Adrian Ulises Mercado Martinez |
| *Asignatura:* | Fundamentos de programación |
| *Grupo:* | 15 |
| *No de Práctica(s):* | 9 |
| *Integrante(s):* | Angeles Estrada Ricardo, Ruiz Godoy Franco, Sampayo Rodríguez Eder, Solorio Garduño Alexis Jair |
| *No. de Equipo de cómputo empleado:* | 37-40 |
| *No. de Lista o Brigada:* | 9 |
| *Semestre:* | 2020-1 |
| *Fecha de entrega:* | 16 de Octubre 2019 |
| *Observaciones:* |  |
| CALIFICACIÓN: |  |

**SALAS A Y B**



**Introducción.**

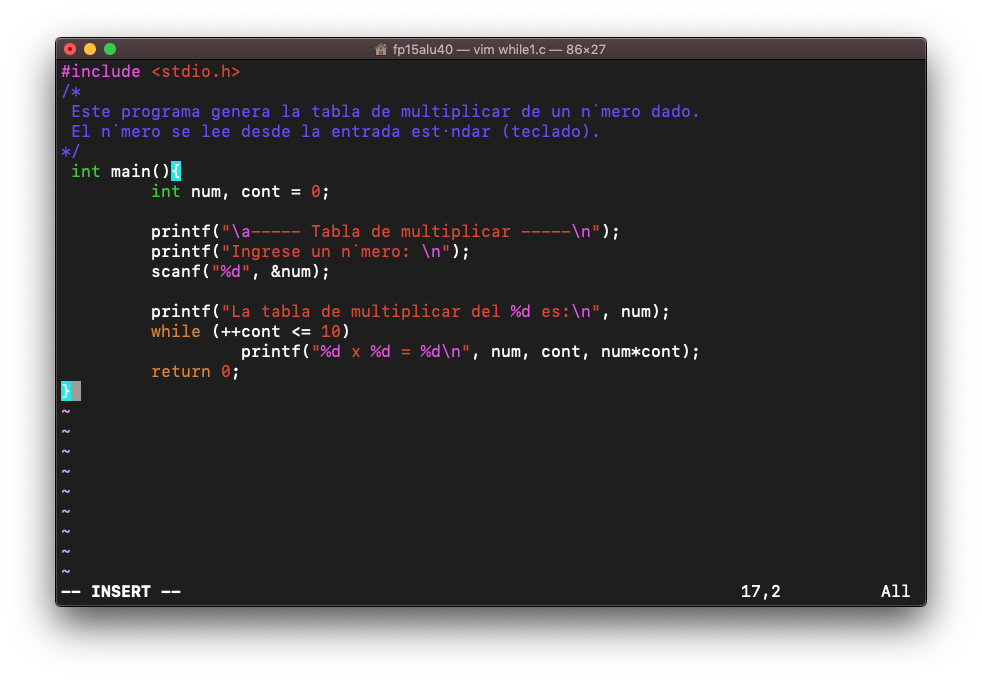
En esta práctica se analizaron varios programas nuevos que utilizan nuevos comandos de estructura de repetición y la directiva define. En lenguaje C existen tres estructuras de repetición: while, do-while y for. Las estructuras while y do-while son estructuras repetitivas de propósito general.

Las estructuras de repetición son las llamadas estructuras cíclicas, iterativas o de bucles. Permiten ejecutar un conjunto de instrucciones de manera repetida (o cíclica) mientras que la expresión lógica a evaluar se cumpla (sea verdadera).

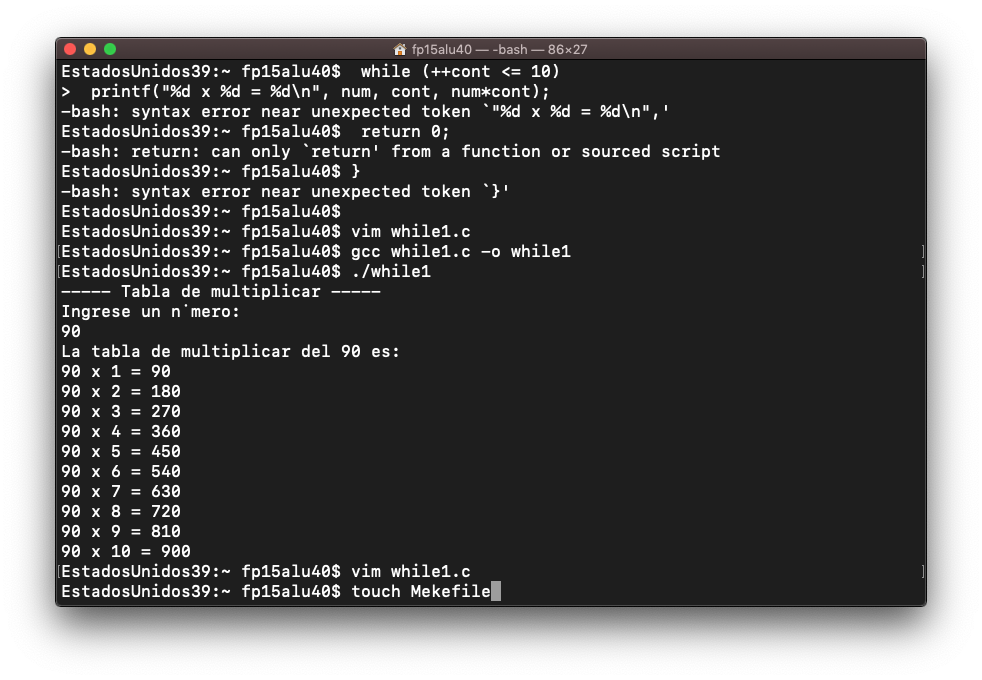
**Desarrollo.**

Lo que vamos a hacer en esta práctica es ver los diferentes estructuras de repetición como “while”, “for”, “do-while” y Define.

Como lo mencionamos anteriormente vamos a ver las diferentes estructuras de repetición, la primera con la que vamos a trabajar va a ser con la estructura “while” e hicimos un programa que lo que hacía era hacer la tabla de multiplicar del número que nosotros le ingresamos.

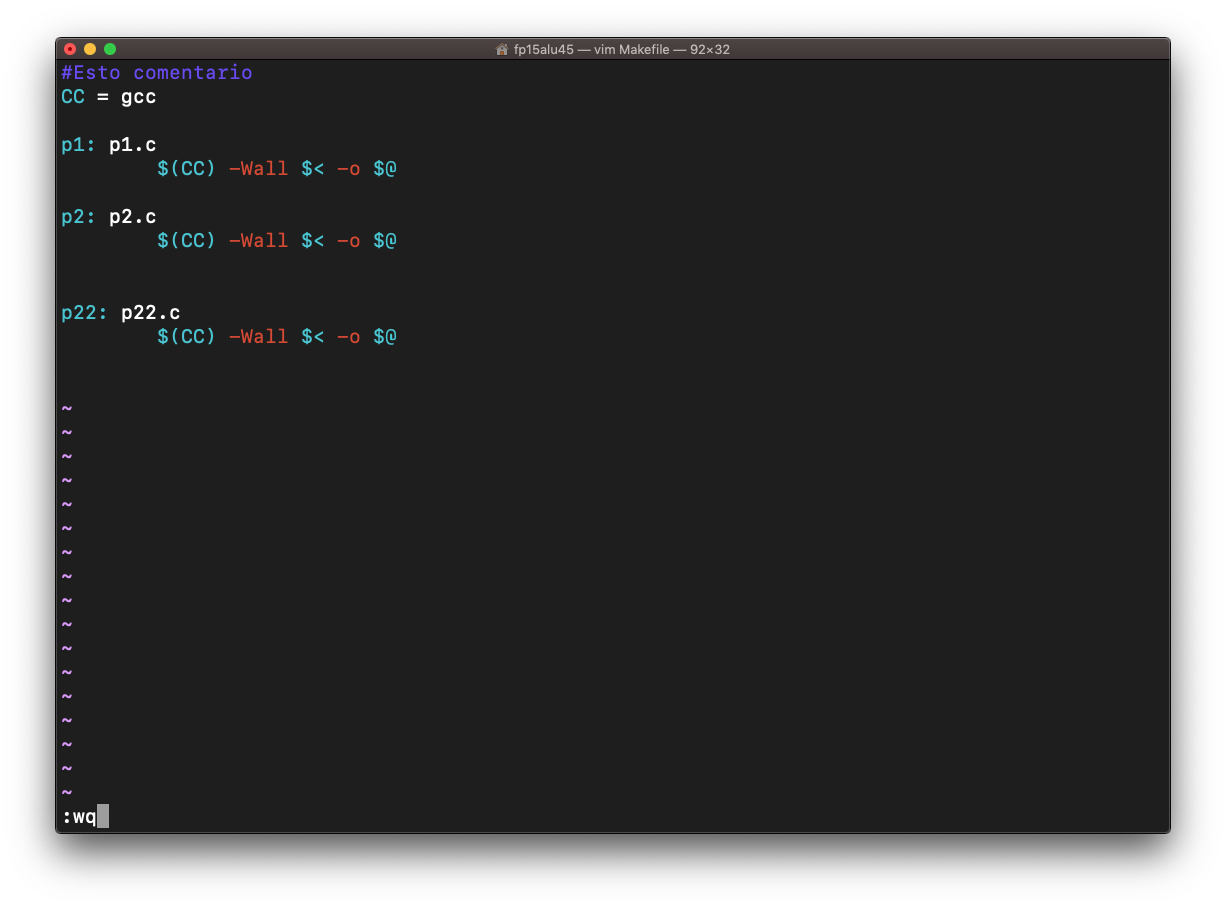
****

Después se compiló y se ejecutó el programa, cabe decir que para este programa utilizamos una manera diferente para compilar el programa “Makefile” lo que hace es mandar a buscar nuestro programa de ejecución ya que dentro de Makefile ya está programada como se llama nuestro archivo y como se llama el archivo de ejecución.

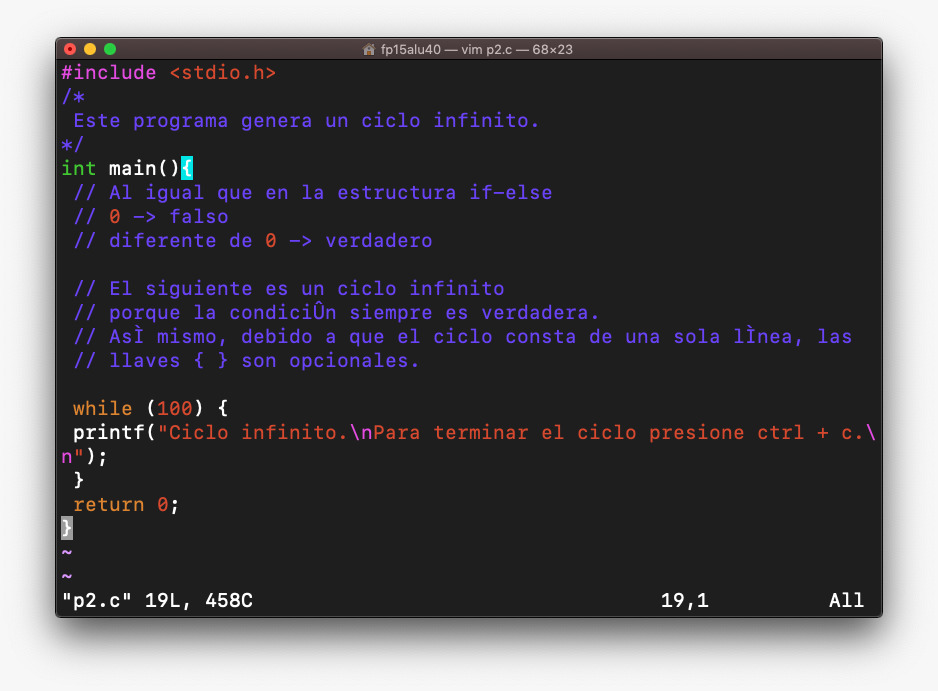
****

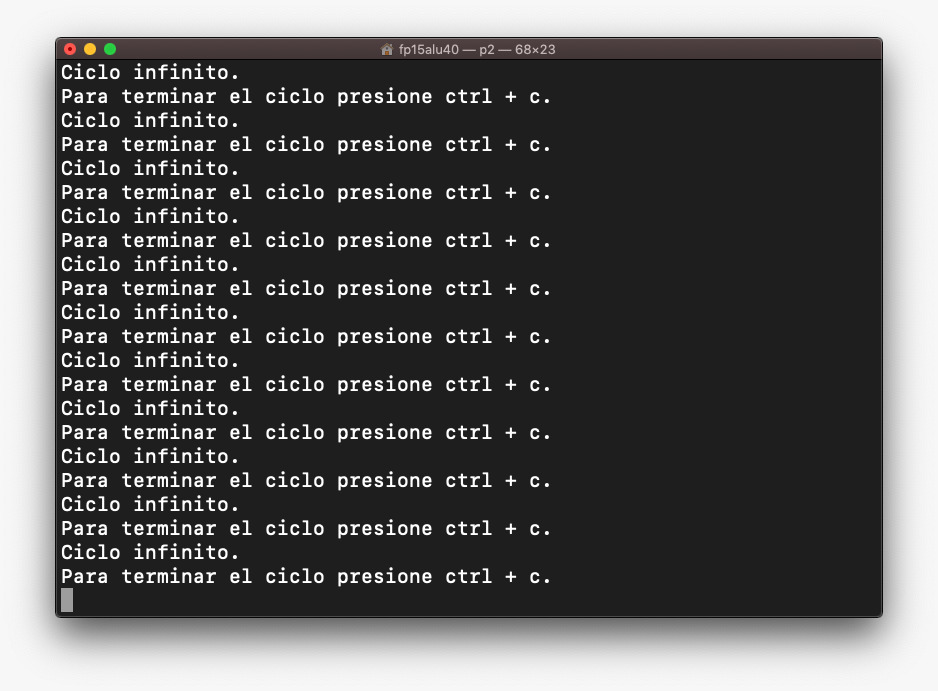
Cabe decir que al principio no podemos compilarlo a través de “Makefile” y lo hacíamos a través de “gcc” y para nuestro primer programa lo compilamos por “gcc”.

Después de ejecutar el programa, a partir de algunas correcciones se pudo compilar los demás archivos mediante “Makefile” lo que hizo que la compilación y ejecución de los demás programas más fácil.

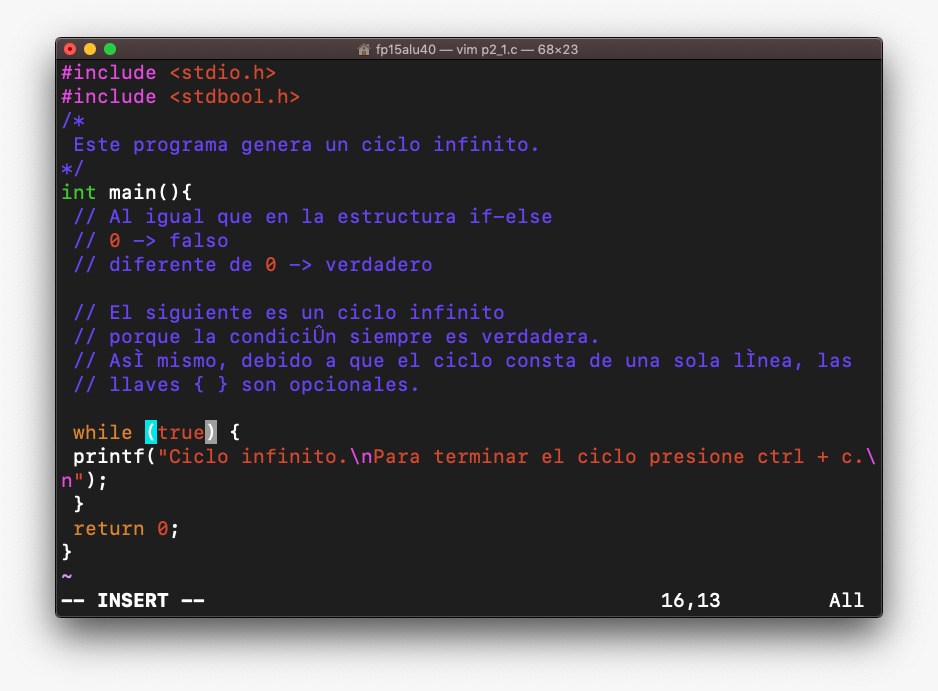


Después utilizamos la misma estructura “while” solo que esta vez se hizo un programa pero este cumplía un ciclo infinito, esto quiere decir que hasta que el programa diera falso iba a parar. Pero en este caso siempre era verdadero por lo que nunca iba a parar hasta que diéramos ctrl + c, ya que esta es una manera para parar de forma forzada el programa. Como se muestra a continuación en las dos imágenes que se encuentran abajo.

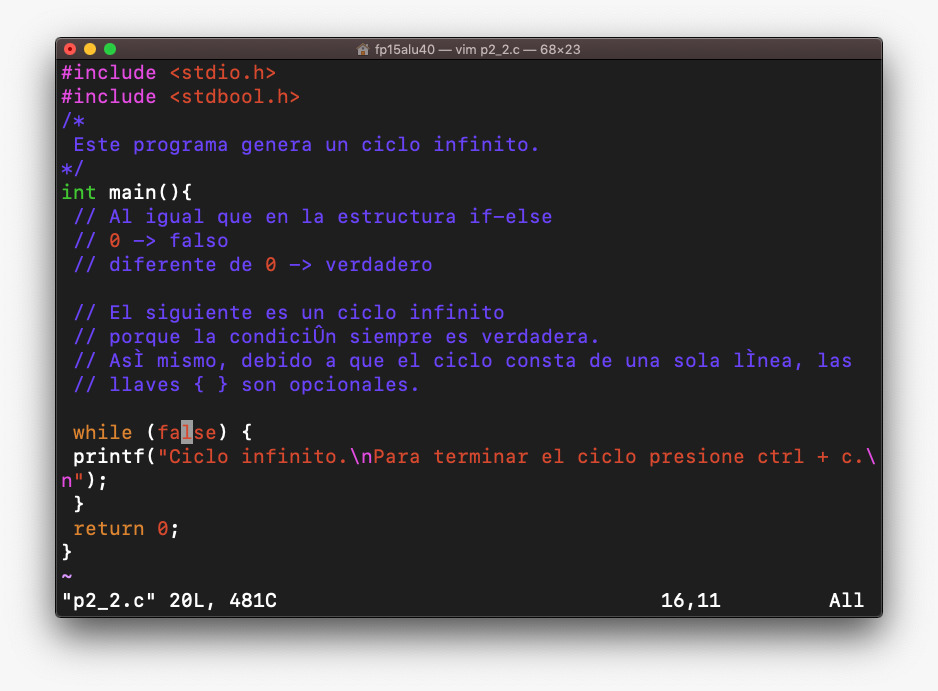




Después en el mismo programa en lugar de poner “100” como en el programa anterior ponemos “true” que es verdadero en inglés, lo que hace exactamente lo mismo que el programa anterior, era un ciclo infinito. Como se muestra a continuación.



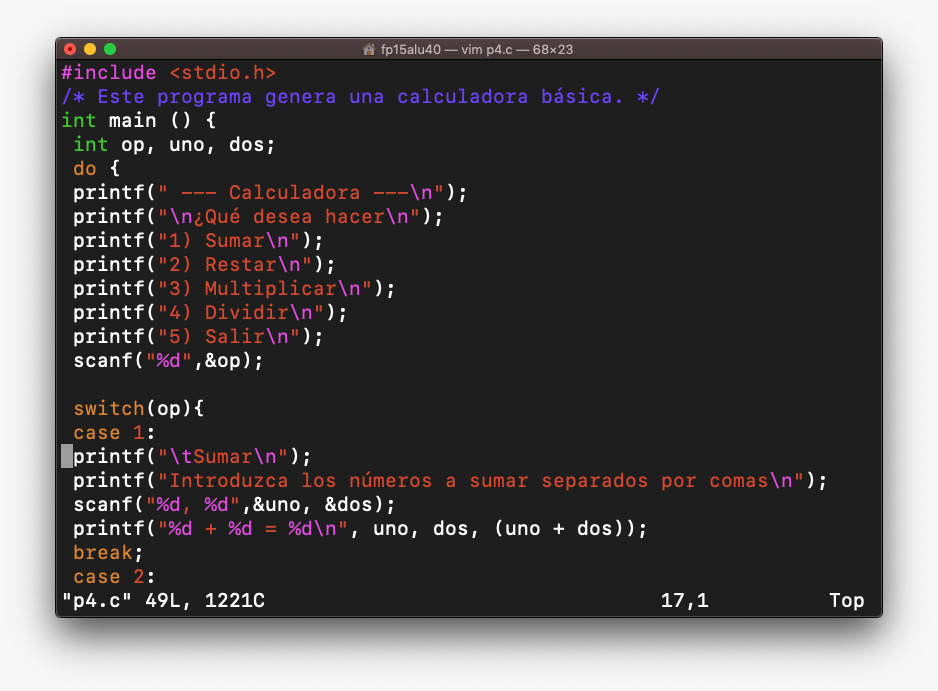
Después volvimos a trabajar en el mismo programa, solo que en lugar de poner “true” ponemos “false” que es falso, lo que hace que este programa no sea un ciclo infinito ya que si es falso se termina el programa.



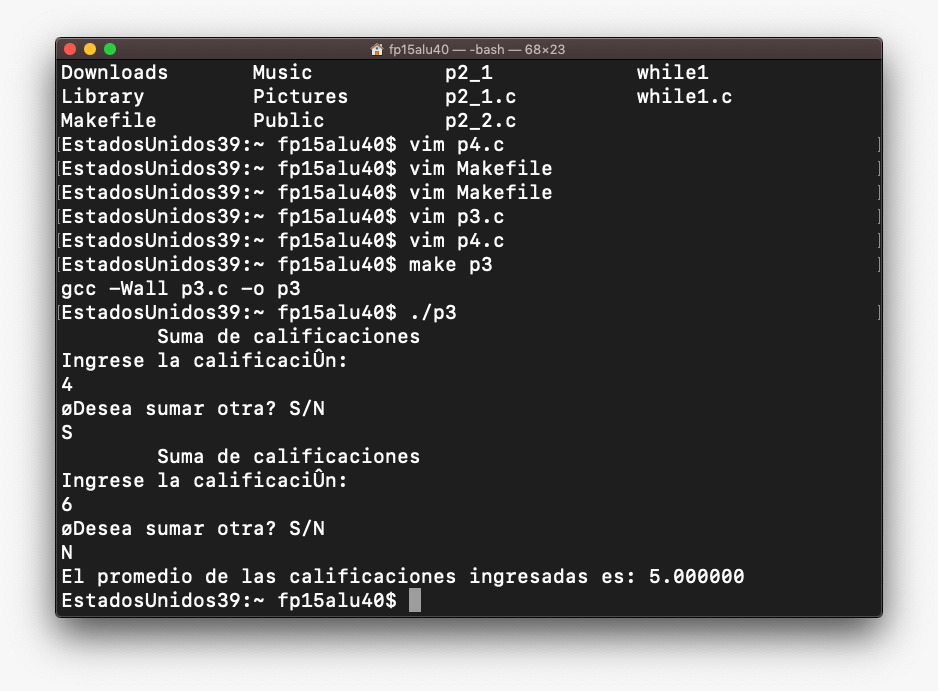
Después utilizamos la estructura “do-while” lo que hacía este programa es ejecutar las instrucciones y después evaluar la condición, y este se repite hasta que la condición de falso.



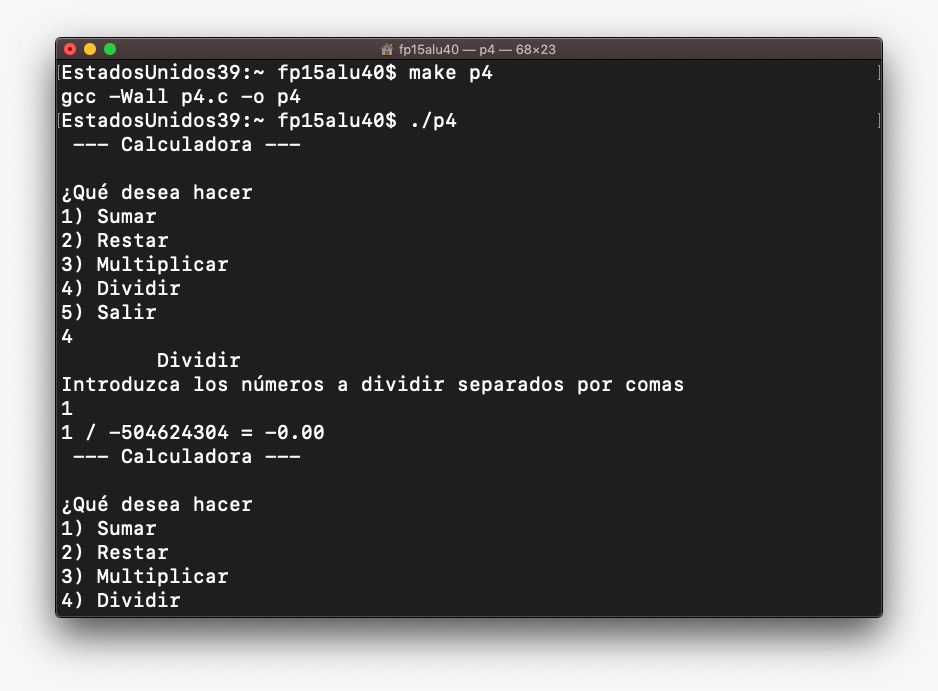
Lo que hacía este programa era pedir las calificaciones y si nosotros queríamos agregar otra teníamos que dar “s” ya que dentro del programa, estaba programada para volver a repetir, pero si ya se quería terminar teníamos que escribir “n” y lo que hacía después era sacar el promedio de todas las calificaciones que nosotros le ingresamos.



Cuando ejecutamos el programa , ya con los datos ingresados, el programa nos mostraba lo siguiente como se muestra en la imagen anterior.

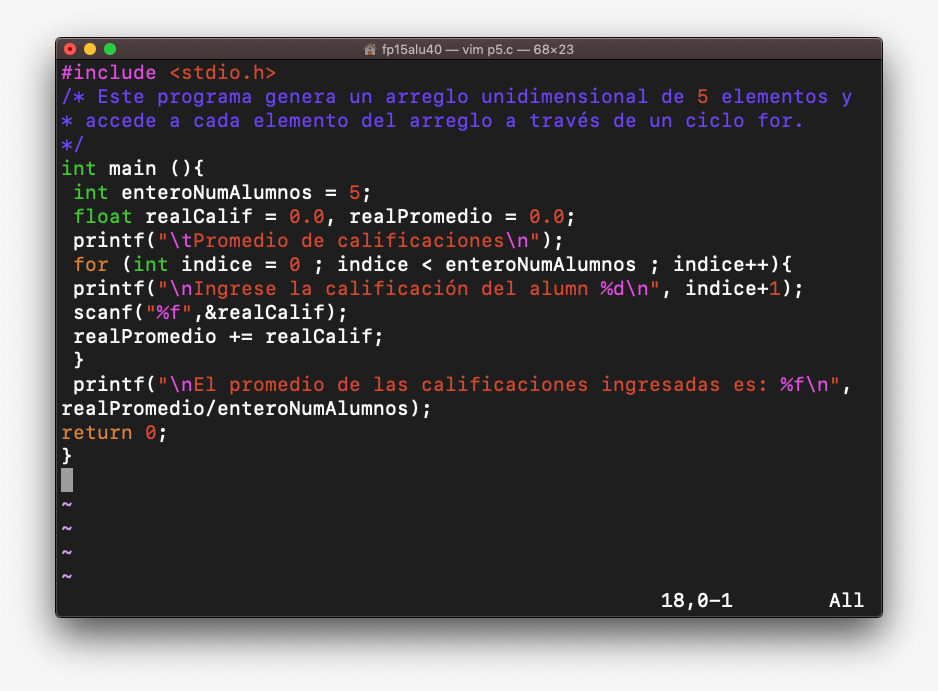


Después realizamos otro programa, igual con la misma estructura “do-while” solo con una pequeña diferencia ya que este programa le agregamos otra estructura, solo que esta es una estructura era selectiva “switch”.



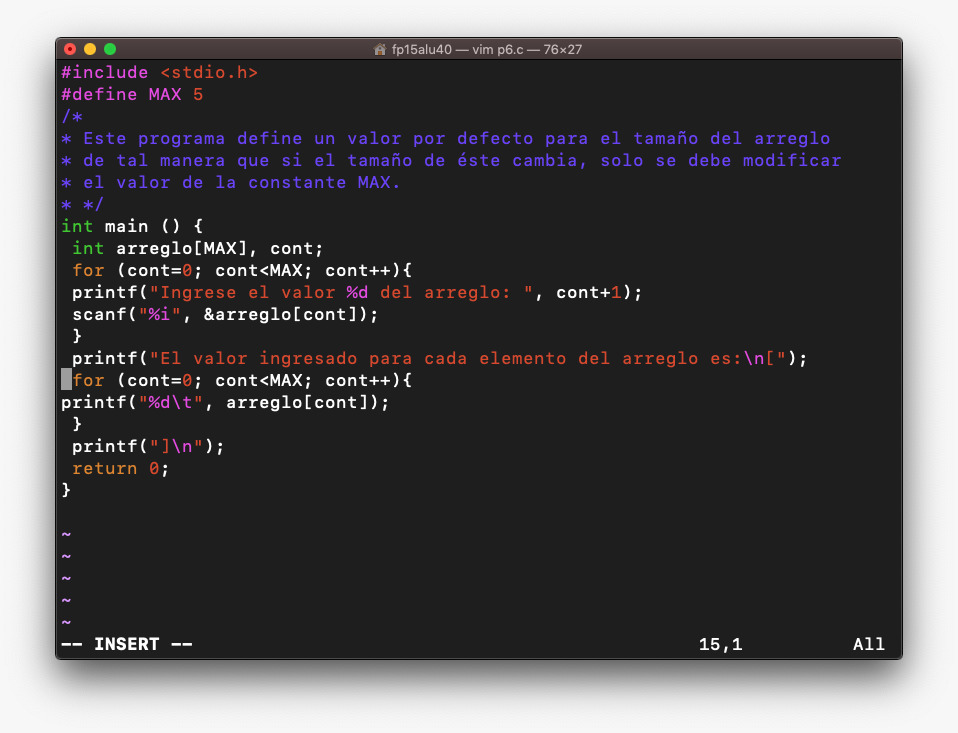
Lo que hace este programa es que dado diferentes datos de entrada, va a realizar una instrucción diferente para cada caso, esto es dentro de “switch” y el “do-while” lo que hace es que ejecuta las acciones una primera vez y después lo evalúa según el dato de entrada ya que si esta es diferente a lo establecido, dará como falso y se acabara el programa.

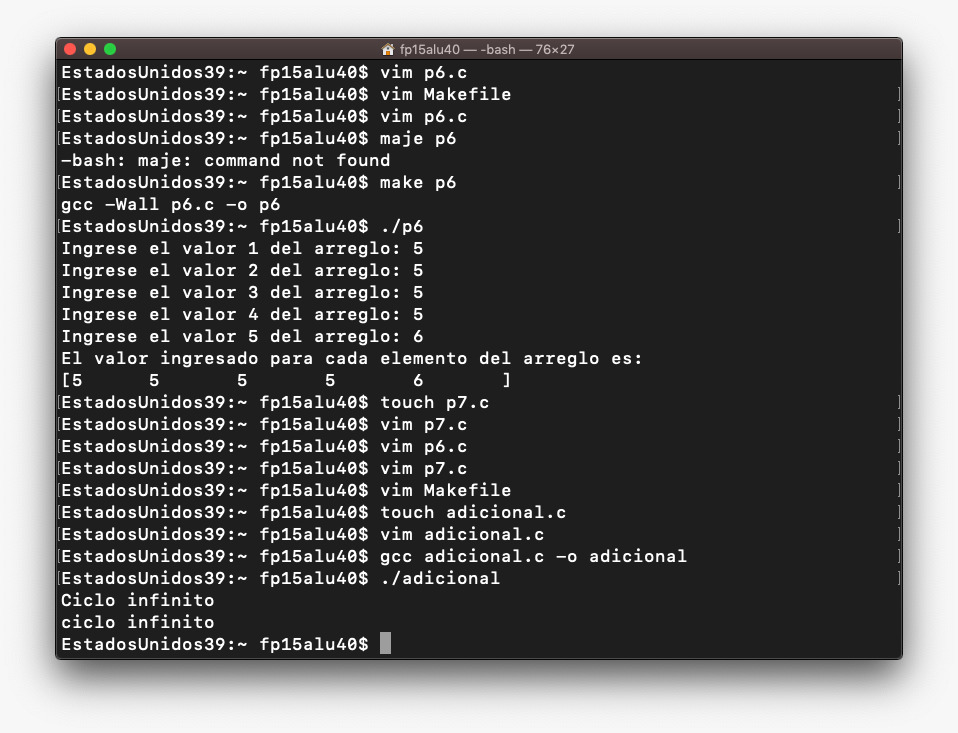
En este programa lo que hace es un promedio de calificaciones de 5 alumnos mediante un arreglo unidimensional, en este arreglo pide las calificaciones de estos “alumnos”.



Dándonos un resultado de este programa el promedio de las calificaciones de estos alumnos.

Después en este programa el propósito de este era de hacer un arreglos, para este se hizo uso de la directiva define, dándonos para este programa un máximo de 5 números en el arreglo.





Compilando por medio de lo que es el “Makefile”, y ejecutándose se obtiene primero que el programa le pide al usuario 5 números para este arreglo, una vez proporcionados se prosigue con el arreglo ordenándolos de mayor a menor los números proporcionados.

Para este programa número 7, se hace uso, de nuevo, de “define” e donde esta vez igual nos pone un límite de ingresar 5 números, y lo que este hace es hacer una suma de estos 5 números ingresados.

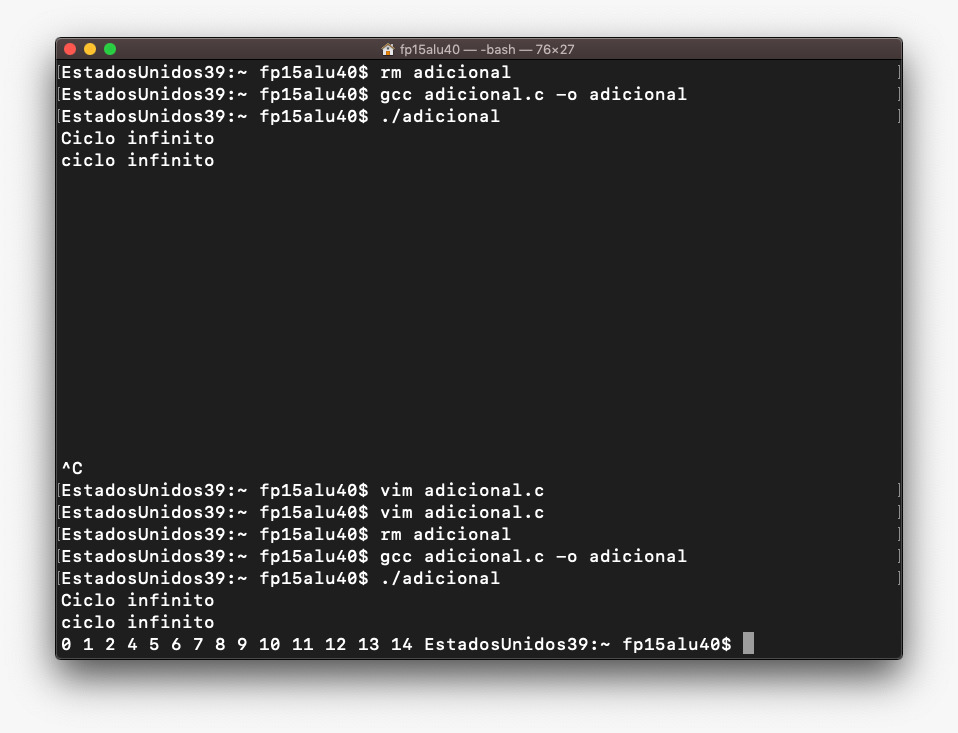


Una vez ingresados los 5 números a calcular hace la suma, pero si los números que se le pidieron al usuario en la suma rebasa la cantidad de 50 este mandara un mensaje diciendo que rebasamos dicho límite, y se detendrá al darse cuenta que lo rebasamos.

Por último se ha hecho un programa adicional donde al hace una secuencia de números desde el cero hasta el 14, ya que aquí le pusimos que fuera menor a 15, en caso de cuando le pones un if y le pusimos que si cont (en este caso nuestra variable contador) fuera igual a 3, pusimos continúe, esto lo que hace es que cuando cont = 3, se brinque ese número.



Arriba se muestra el programa y a continuación se mostrará lo que hace el programa a la hora de compilarlo, como podemos ver escribe todos los números del 0 al 14, todos menos el 3.



**Conclusiones.**

En esta práctica se vio las diferentes estructuras cíclicas, las cuales como ya vimos, nos sirven mucho para cunado una variable puede tomar varios valores y dependiendo del valor este haga un proceso, esta práctica ayuda a la resolución de problemas complejos y son de las estructuras que más se usan en el lenguaje C, lo malo de estos lenguajes de programación es que si te compila muchas veces no hace lo que quieres que hago, y si no compila puede ser por diversas razones, pero estas estructuras ayudan también a oragnizar la información y poder ver mejor el programa para su análisis y corrección.

Angeles Estrada Ricardo.

Finalmente al término de la práctica se ha entendido las estructuras de selección y lo que es la "directiva" define, ya que nos muestra esta práctica diferentes ejemplos de éstos, se puede decir que estas estructuras son útiles y facilitan varios trabajos, pero se debe tener cierto cuidado al escribir el código, ya que, se puede llegar a generar problemas de cierre de procesos, en tanto a "define" se ha entendido su uso y como es aplicada durante se está programando en C.

Ruiz Godoy Franco.

Como conclusión podemos decir que esta práctica alcanzamos todos nuestros objetivos ya que realizamos diferentes programas utilizando las diferentes estructuras de repetición, lo que hace que al momento de programar en C con ayuda de estas estructuras de repetición se nos puede facilitar ciertos problemas a los que nos tengamos que enfrentar.

Solorio Garduño Alexis Jair