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

Laboratorios de computación

salas A y B

|  |  |
| --- | --- |
| *Profesor:* | RODRIGUES ESPINO CLAUDIA |
| *Asignatura:* | FUNDAMENTOS DE PROGRAMACION |
| *Grupo:* | 4 |
| *No de Práctica(s):* | PRACTICA 11: Arreglos unidimensionales y multidimensionales. |
| *Integrante(s):* | FLORES MENDOZA OLGA |
|  |  |
| *No. de Equipo de cómputo empleado* | 44 |
| *Semestre:* | 2019-2 |
| *Fecha de entrega:* | 4 DE MAYO DE 2019 |
| *Obervaciones:* |  |
|  |  |

COMPILACION

CALIFICACIÓN: \_\_\_\_\_\_\_\_\_\_

**PRACTICA 11**

**OBJETIVOS:**

* Reconocer la importancia y utilidad de los arreglos, en la elaboración de programas que resuelvan problemas que requieran agrupar datos del mismo tipo, así como trabajar con arreglos tanto unidimensionales como multidimensionales.

**ACTIVIDADES:**

* Elaborar un programa en lenguaje C que emplee arreglos de una dimensión. 
* Resolver un problema que requiera el uso de un arreglo de dos dimensiones, a través de un programa en lenguaje C. 
* Manipular arreglos a través de índices y apuntadores.

**INTRODUCCION:**

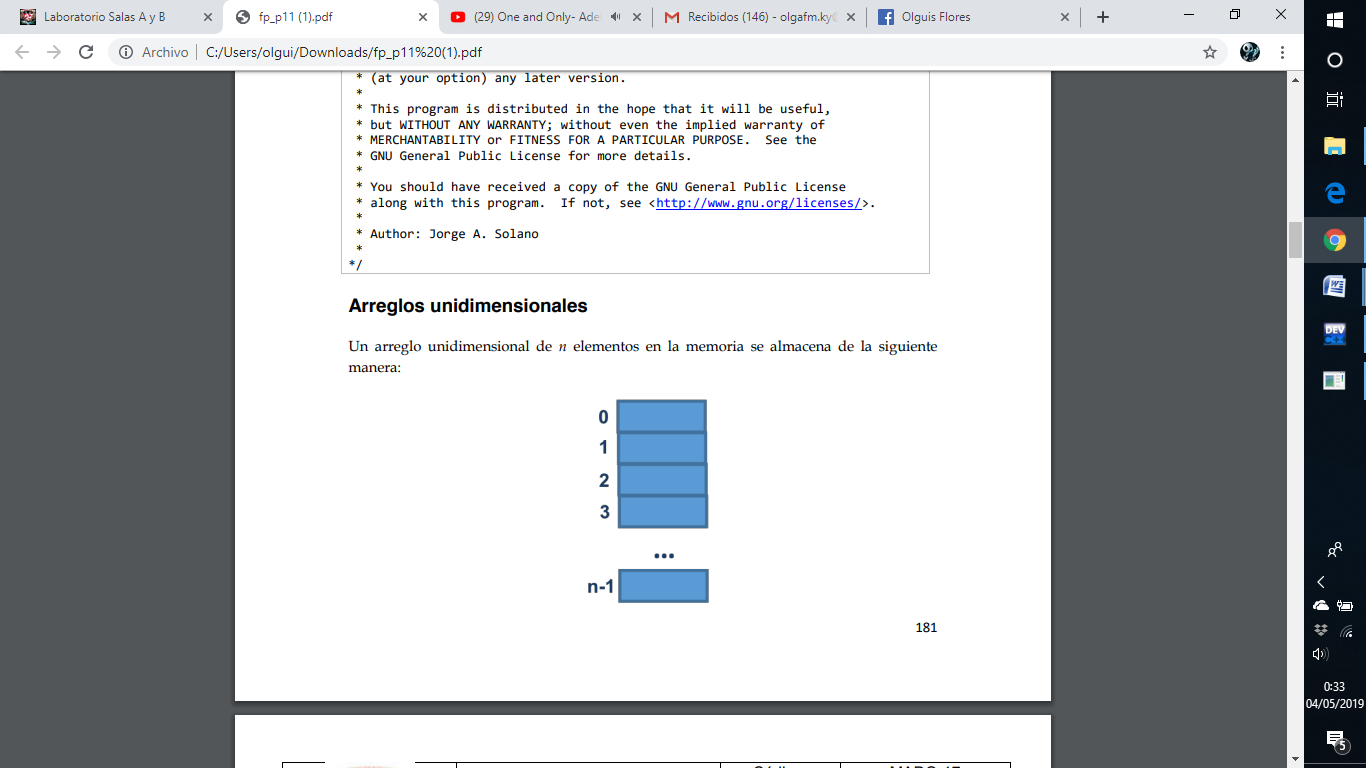
Un arreglo es un conjunto de datos contiguos del mismo tipo con un tamaño fijo definido al momento de crearse. A cada elemento (dato) del arreglo se le asocia una posición particular, el cual se requiere indicar para acceder a un elemento en específico. Esto se logra a través del uso de índices.

**¿Qué son los puntadores?** Un apuntador es una variable que contiene la dirección de una variable, es decir, hace referencia a la localidad de memoria de otra variable. Debido a que los apuntadores trabajan directamente con la memoria, a través de ellos se accede con rapidez a un dato. La sintaxis para declarar un apuntador es:

**TipoDeDato \*apuntador, variable; apuntador = &variable;**

Tipos de arreglos:

* **Unidimensionales:** De n elementos en la memoria se almacena de la siguiente manera;

La primera localidad del arreglo corresponde al índice 0 y la última corresponde al índice n-1, donde n es el tamaño del arreglo. La sintaxis para definir un arreglo en lenguaje C es la siguiente:

**tipoDeDato nombre[tamaño]**

Donde nombre se refiere al identificador del arreglo, tamaño es un número entero y define el número máximo de elementos que puede contener el arreglo. Un arreglo puede ser de los tipos de dato entero, real, carácter o estructura.

* **Multidimensionales**: Lenguaje C permite crear arreglos de varias dimensiones con la siguiente sintaxis:

**tipoDato nombre[ tamaño ][ tamaño ]…[tamaño];**

El nombre y tamaño se definieron en el punto anterior (el número de dimensiones está determinado por el número de corchetes). De manera práctica se puede considerar que la primera dimensión corresponde a los renglones, la segunda a las columnas, la tercera al plano, y así sucesivamente.

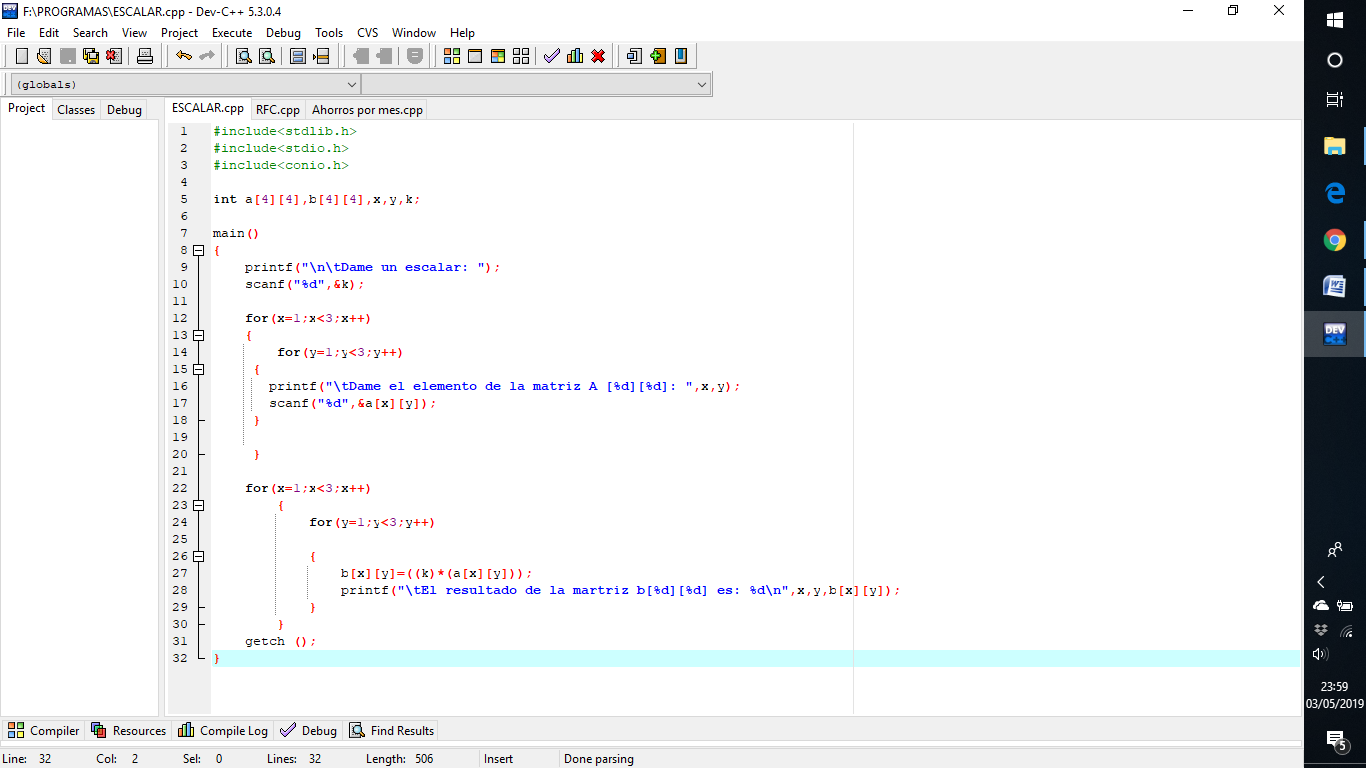
Los arreglos se utilizan para hacer más eficiente el código de un programa.

**DESARROLLO DE ACTIVIDADES:**

Los siguientes programas fueron realizados en Dev-C++ empleando arreglos y apuntadores.

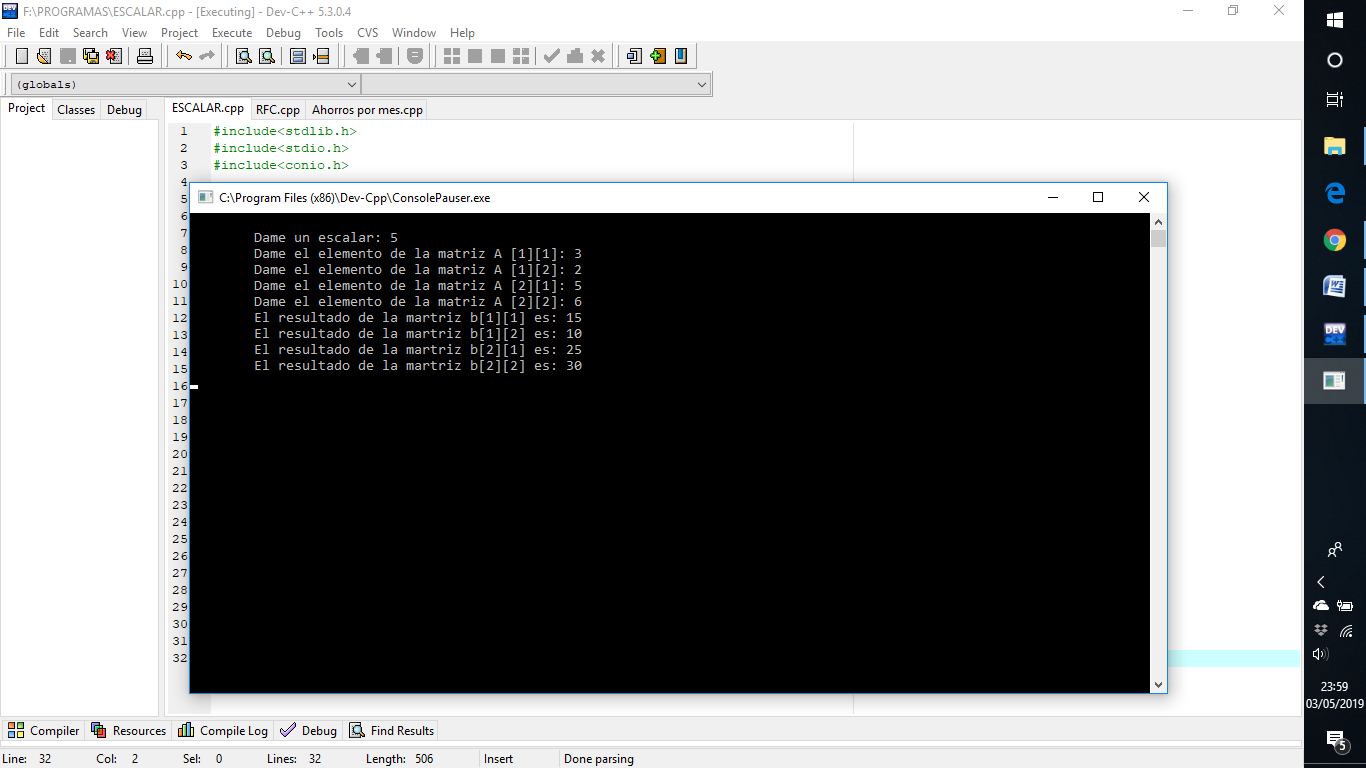
**PROGRAMA 1:** Este programa multiplica una matriz por un escalar. Ambos datos están dados por el usuario.

Para este programa utilizamos dos arreglos y la estructura de repetición for.

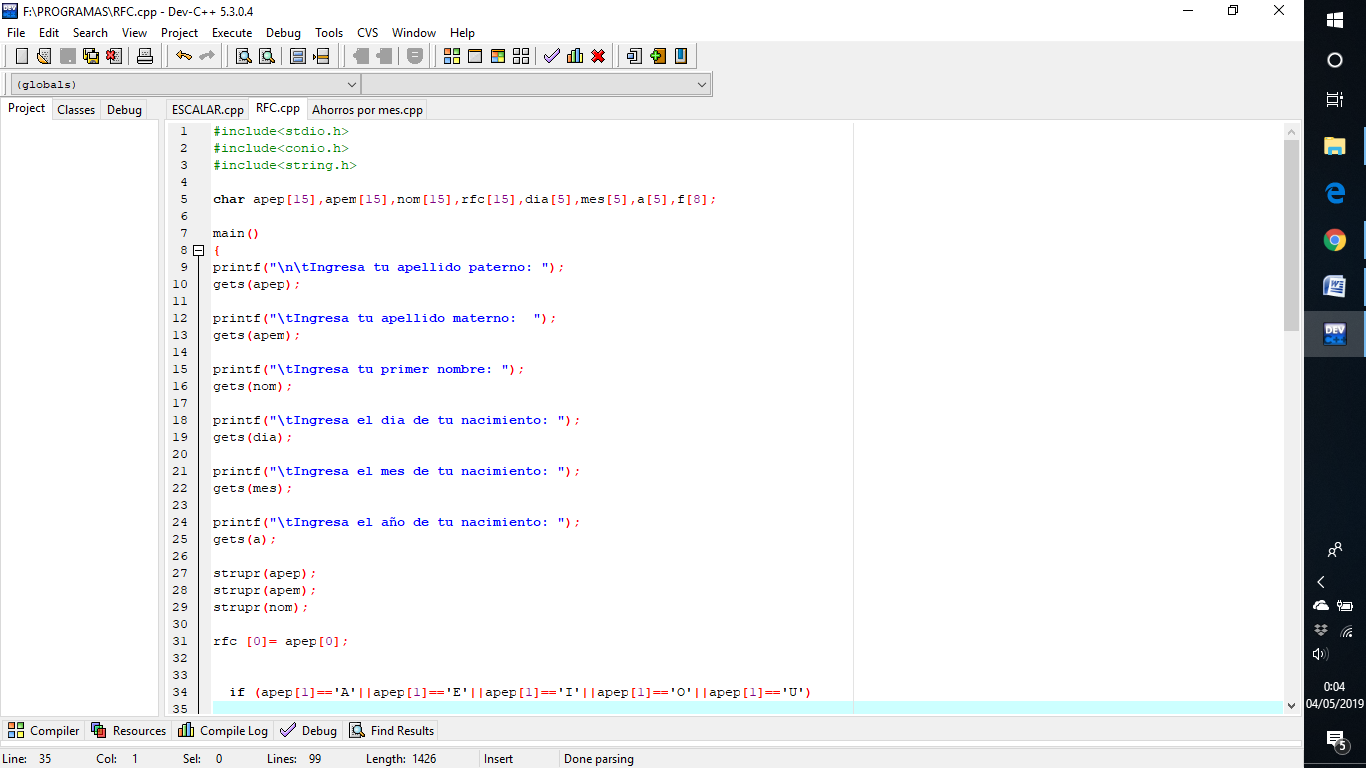
****

ARREGLO

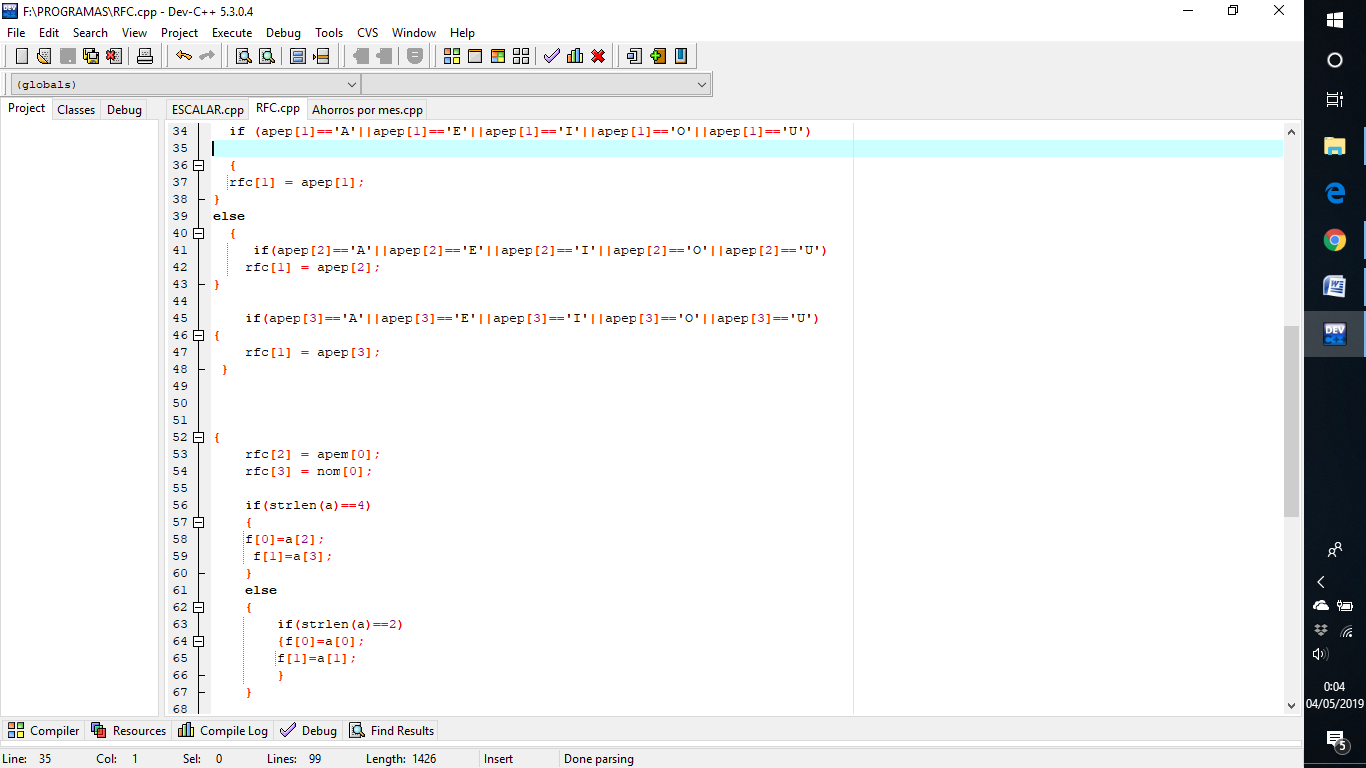
Compilamos y se procedió a la ejecución del programa. El cual corrió de manera exitosa.

****

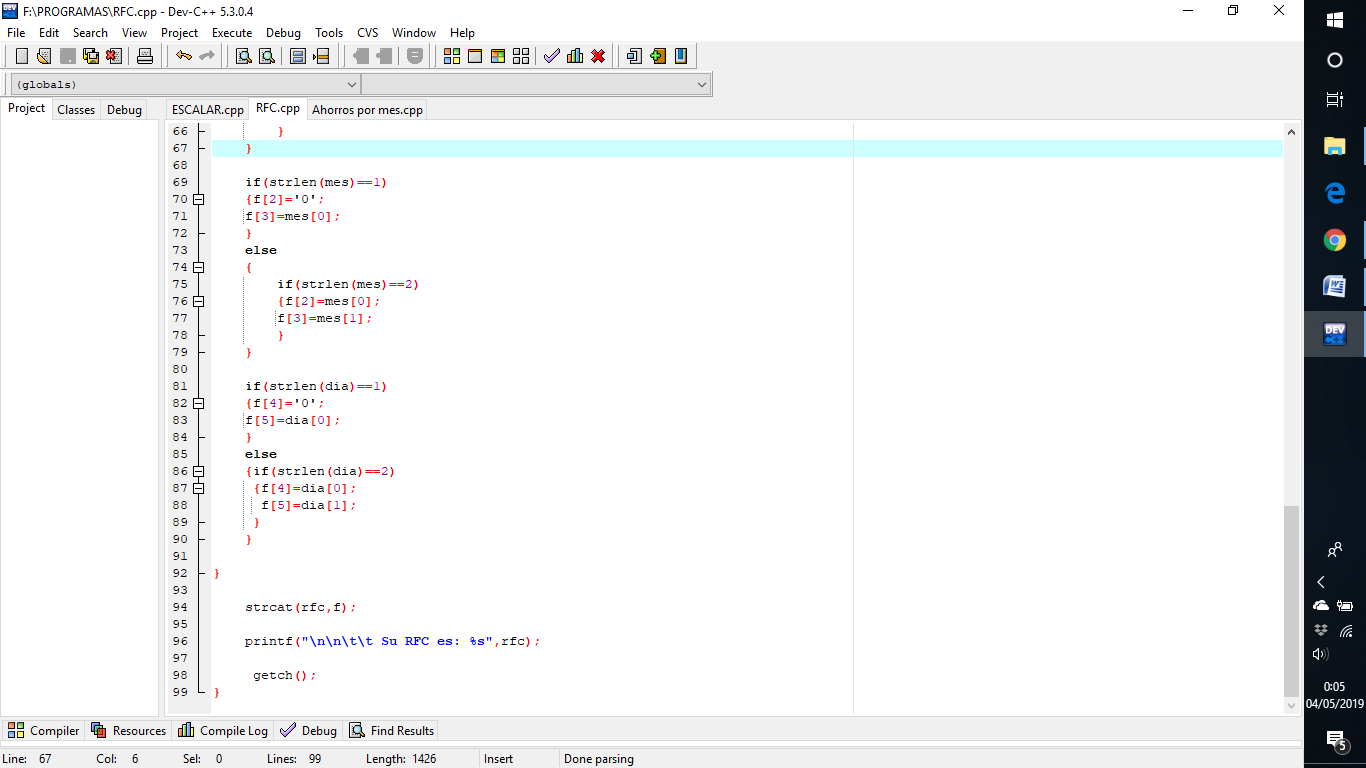
**PROGRAMA 2:** El siguiente programa obtiene el RFC del usuario con base en los datos proporcionados.

Además de usar arreglos, que en este caso fueron 8, se emplearon las estructuras selección if e if-else. ****

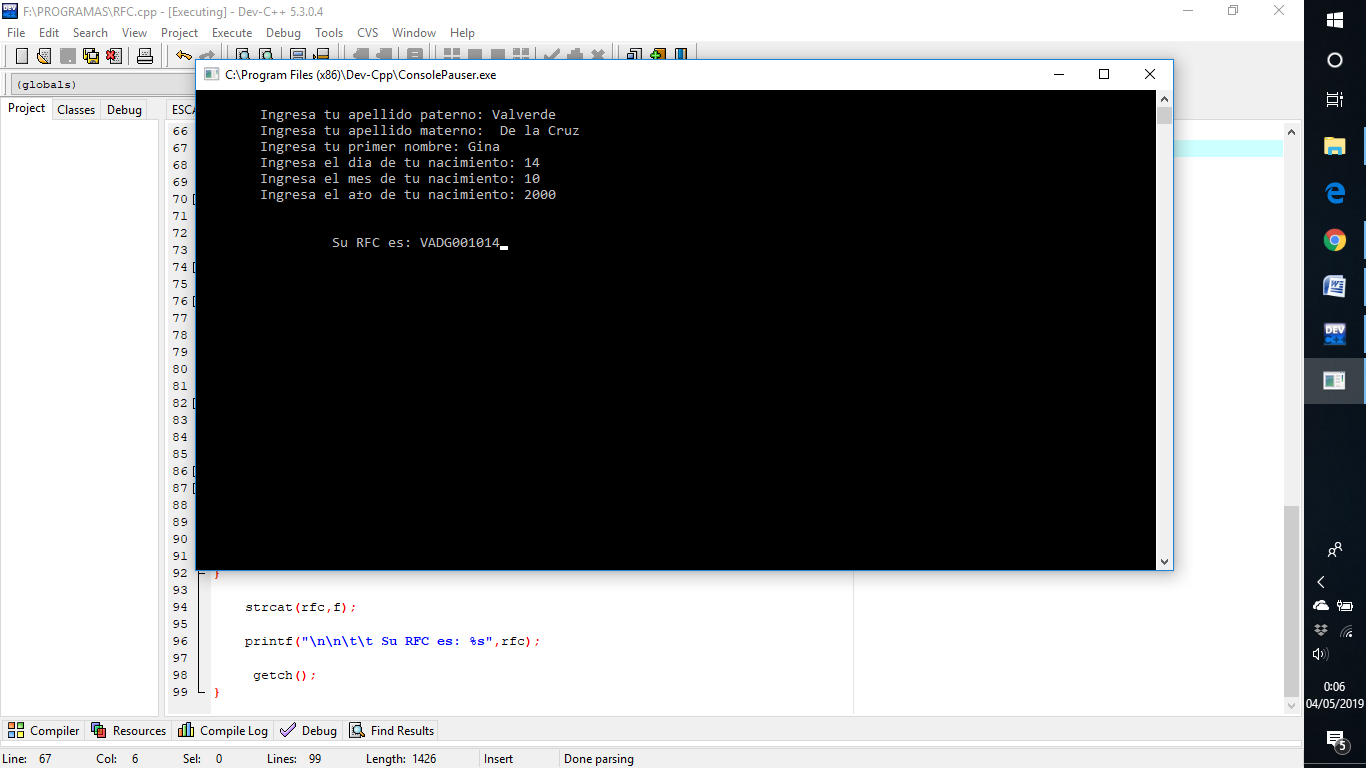
ARREGLOS



Estructura de selección.

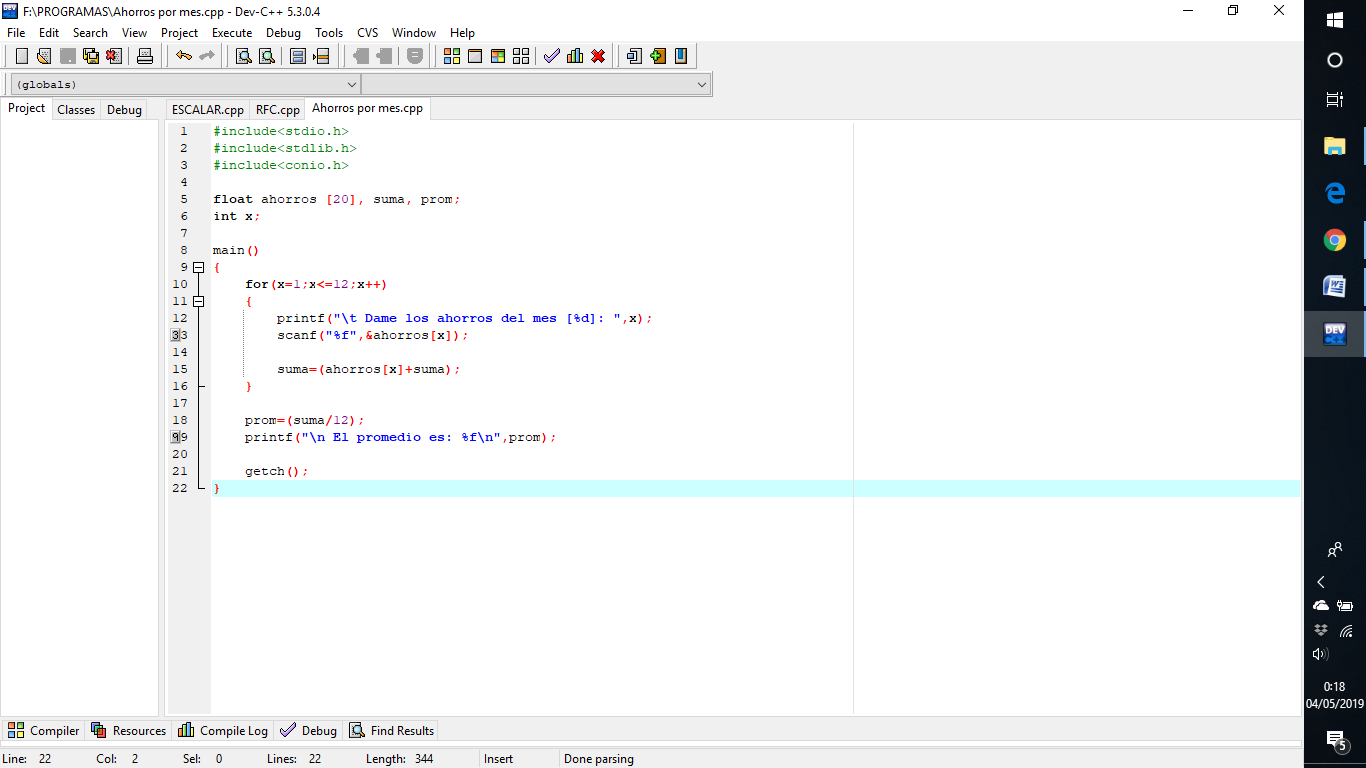
****

Se compilo y ejecuto el programa para verificar si funcionaba correctamente.

****

**PROGRAMA 3:** Este programa guarda los ahorros por mes, los cuales son ingresados por el usuario y los imprime en forma de lista. Posteriormente obtiene el promedio de los valores dados.

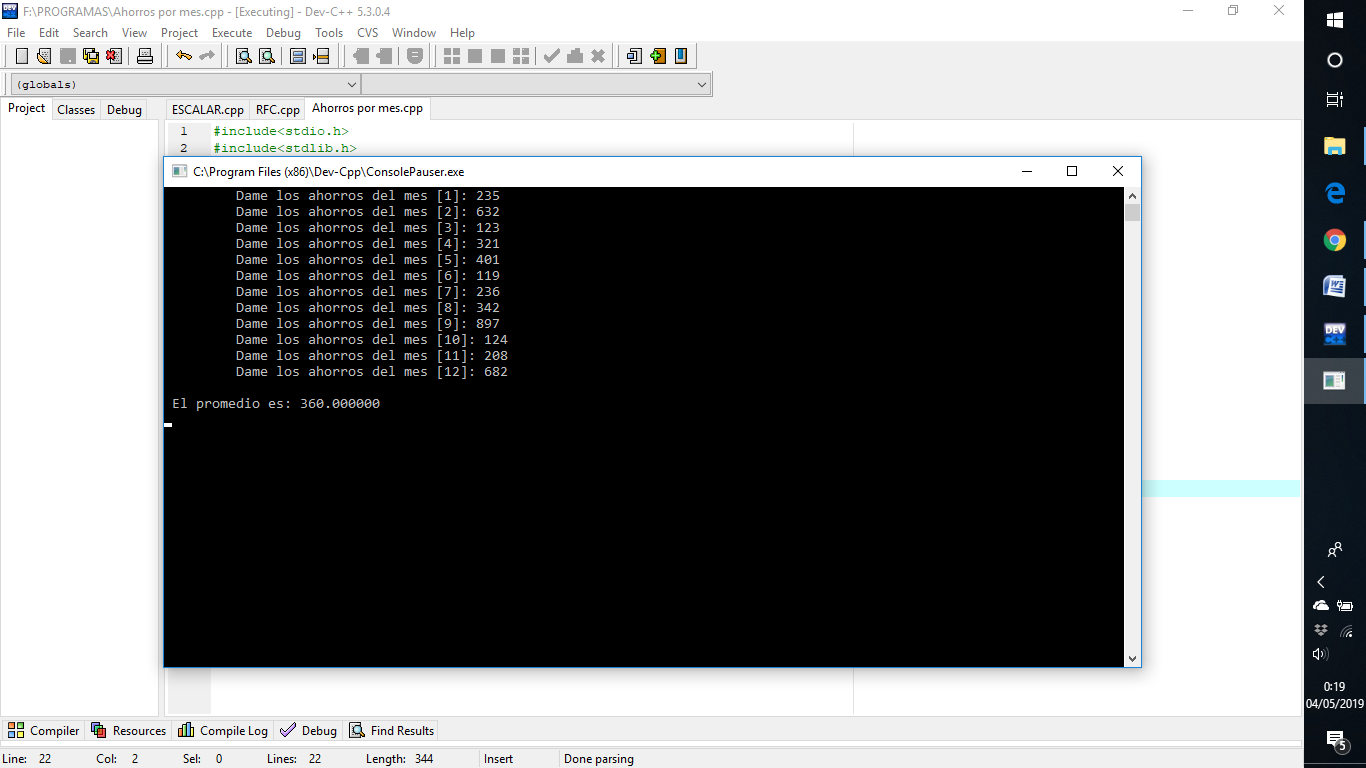
Aquí se hizo uso de un arreglo y un for.



ARREGLO

El programa fue compilado, lo errores que se presentaron fueron corregidos y por último se ejecuto.

Así se verifico si marchaba bien.



**CONCLUSIONES:**

Los arreglos son difíciles de entender al principio, sobre todo porque tiene un nuevo modo de emplearse. Se debe ser cuidadoso en la manera que son declarados. En sus parámetros es conveniente dejar un lapso grande de espacios como lo hicimos en el programa del RFC, de esta manera se da oportunidad al usuario de meter un nombre o apellido que contenga varias letras y esto no será problema. Esta estructura es de gran ayuda para la elaboración de programas más complejos, ya que redice el tiempo de líneas y de tiempo. No importa el nombre que se le dé al arreglo pero es importante recordarlo y proponer unos parámetros convenientes.