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

Laboratorios de computación

salas A y B

|  |  |
| --- | --- |
| *Profesor:* | Claudia Rodriguez Espino |
| *Asignatura:* | Fundamentos de programación |
| *Grupo:* | 1102 |
| *No de Práctica(s):* | 10 |
| *Integrante(s):* | García León César André |
| *Semestre:* | Primero |
| *Fecha de entrega:* | Viernes 27 Octubre 2017 |
| *Observaciones:* |  |
|  |  |

CALIFICACIÓN: \_\_\_\_\_\_\_\_\_\_

Guía práctica de estudio 10:

Depuración de programas

Objetivo:

Aprender las técnicas básicas de depuración de programas en C para revisar de manera precisa el flujo de ejecución de un programa y el valor de las variables; en su caso, corregir posibles errores.

Desarrollo:

En ésta práctica se aprendió lo referente a la depuración de programas que es la acción en la que un programa se ejecuta por medio de herramientas especializadas que nos permiten ver el flujo que va tomando el proyecto a lo largo de su desarrollo, además es importante porque se observa el cambio de valores que van tomando las variables para que de ésta manera uno pueda identificar los errores que tiene el respectivo proyecto. Se analizaron algunos conceptos que son importantes en el desarrollo del software tales como son:

***Error:*** Acción humana que genera un resultado incorrecto

***Defecto (fault):*** Se refiere a un error en el software y es encontrado porque genera una falla.

***Falla (failure):*** Es una desviación del resultado que se espera.

Después se vio lo útil que puede ser un depurador en el campo de la programación, ya que se ocupa para acciones como la optimización de un programa, verificar dónde se encuentra una posible falla que evita que se muestren los resultados esperados, ver algún error de ejecución que es cuando se detiene un programa inesperadamente, etc…

Posteriormente se estudiaron las funciones básicas que tiene un determinado depurador que son:

***Ejecutar un programa***: Es donde se permite ejecutar el programa desde la herramienta de depuración.

***Mostrar el código fuente:*** Es donde se muestra el código fuente e indica la parte del código que se está ejecutando cuando se corre el programa.

***Punto de ruptura:*** Detiene la ejecución de un programa en un determinado punto seleccionado por el programador y se utiliza para ver el flujo del programa, también se pueden observar los cambios de valores que sufren las variables y además es fundamental ya que se permite visualizar hasta qué punto el programa corre adecuadamente y en qué momento el programa tiene un fallo que hace que el programa se ejecute de manera inadecuada.

***Continuar:*** Es la que permite continuar la ejecución de un programa después de haber utilizado un punto de ruptura.

***Ejecutar la siguiente instrucción:*** Permite después de haber detenido el programa ejecutar una instrucción más y volver a detener el programa, ésta acción es útil si se desea analizar a detalle una parte del programa ya que se puede ver lo que hace el programa paso a paso.

***Ejecutar la siguiente línea:*** Hace todas las instrucciones necesarias para llegar a la siguiente línea del código.

***Ejecutar la instrucción o línea anterior:*** Deshace los efectos hechos por alguna otra función a fin de repetir una determinada parte del programa.

***Visualizar el valor de las variables:*** Ésta parte es una de las más importantes ya que se muestra el cambio que van tomando las variables en el flujo del programa y permite detectar los errores que hacen que el programa no arroje o realice lo que se desea obtener.

Por último, se explicó la depuración que realiza dev c++ que es el editor que ocupamos para programar en C siguiendo paso a paso cómo se utiliza.

Primero se debe tener el código fuente escrito en el editor, después en el menú de herramientas se habilitó la opción de **“*generación de información de depuración”***haciendo que de ésta manera el programa pueda compilarse, depurarse y ejecutarse.

Los puntos de ruptura se obtienen al dar clic en una línea del código, ésta línea se volverá de color rojo, si se desea quitar se vuelve a dar clic en la línea correspondiente. Para depurar un programa primero debe compilarse, posteriormente se depura haciendo que se abra una barra de herramientas correspondiente a la depuración, el punto de ruptura en este momento se tornará color azul.

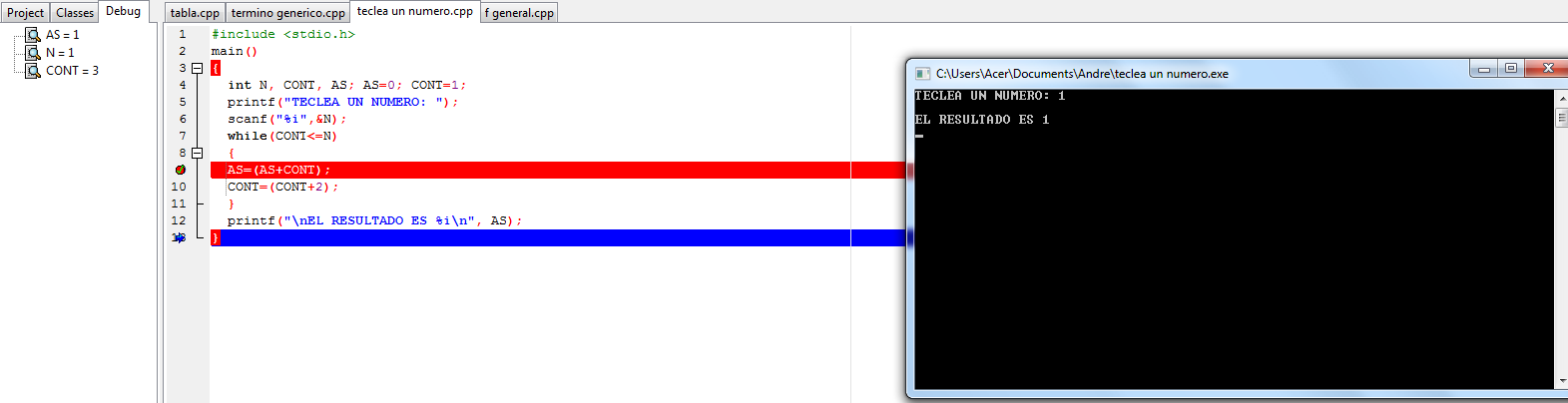
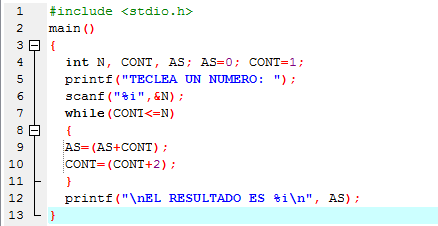
Entre las herramientas se encuentra **“*siguiente paso”***que es el que ejecuta la línea siguiente; la función ***“avanzar paso a paso”*** ejecuta instrucción por instrucción,

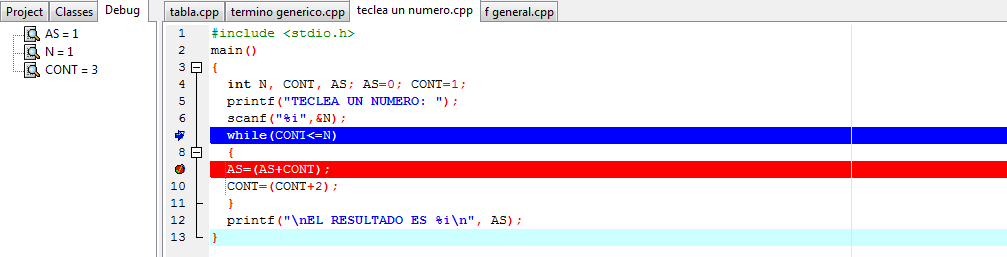
***“saltar paso”*** ejecuta hasta que se encuentre otro punto de ruptura, si se desea detener la depuración se oprime ***“parar ejecución”*** y por último para estudiar el valor de las variables se agregan con la función ***“añadir watch”*** en donde se agregan los nombres de las variables, así se abre una ventana a la izquierda en donde se muestra el cambio de valor de cada variable respectivamente.

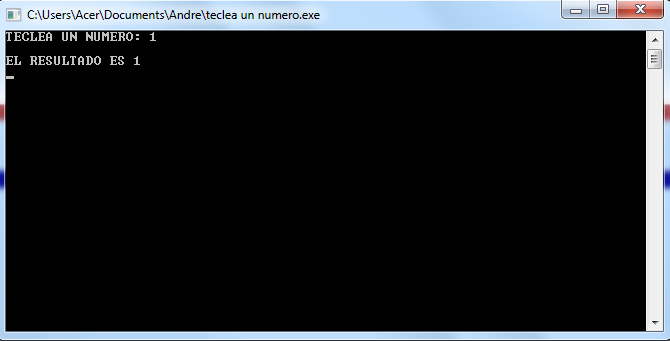
ACTIVIDADES A REALIZAR EN LA PRÁCTICA

***Depuración de los siguientes programas:***

Programa valor de una variable AS

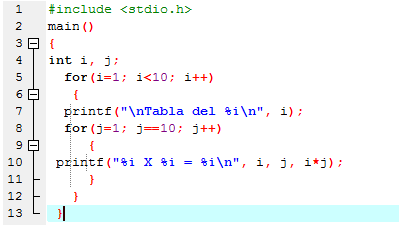




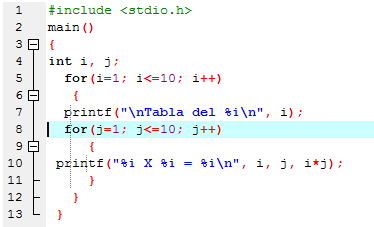


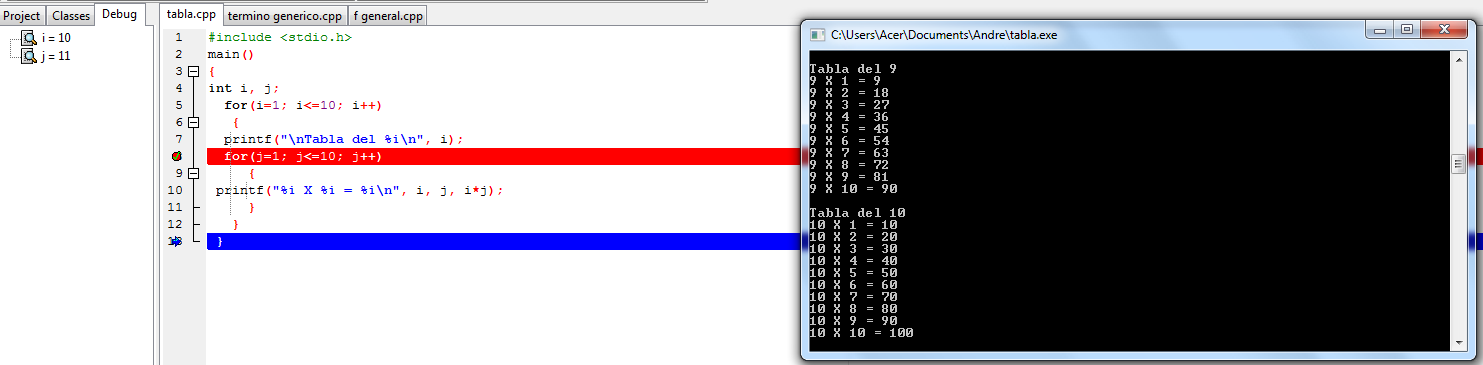
Programa tablas del 1 al 10

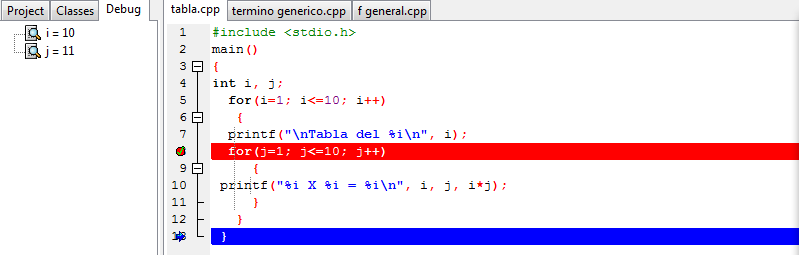
Programa original

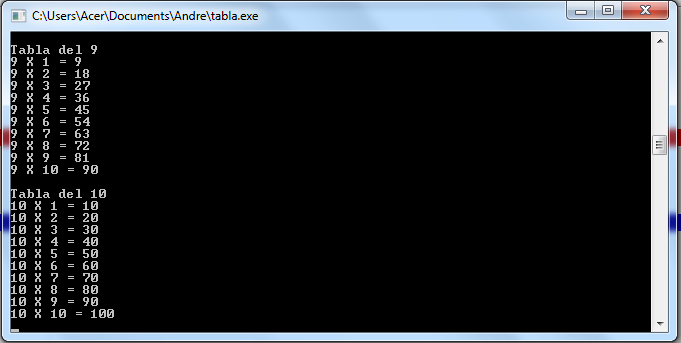


Programa depurado



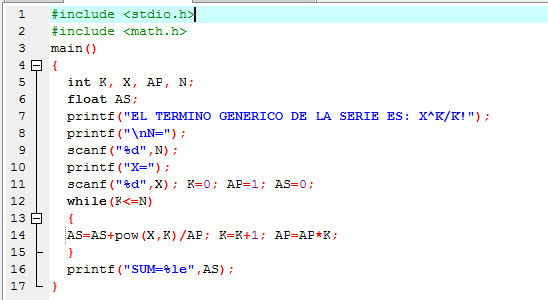




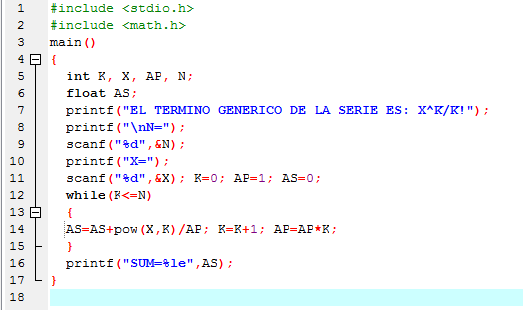


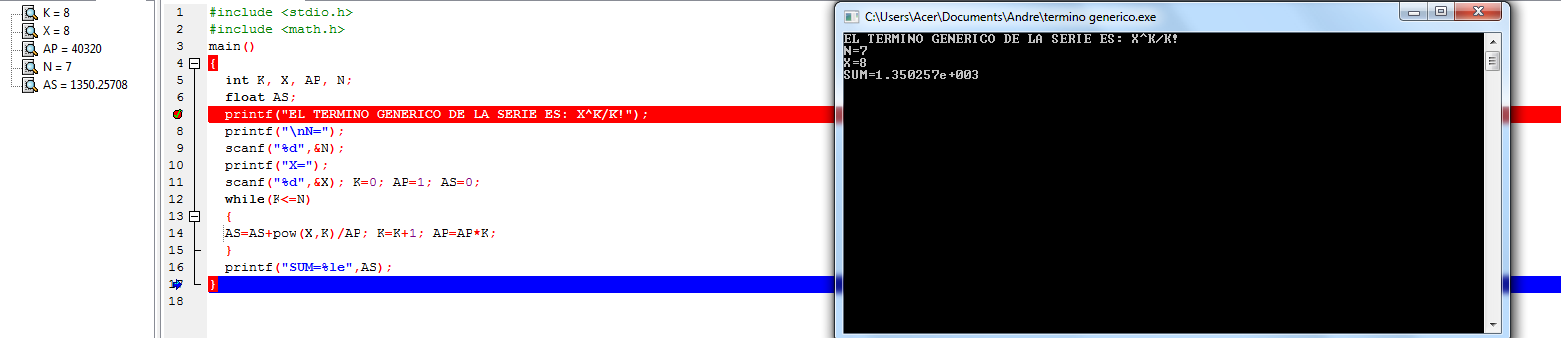
Programa término genérico

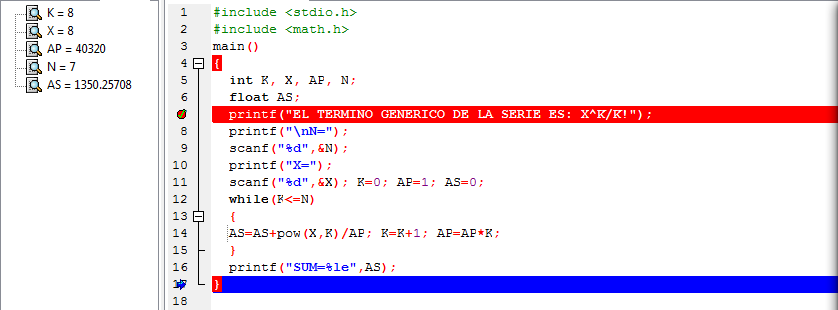
Programa original

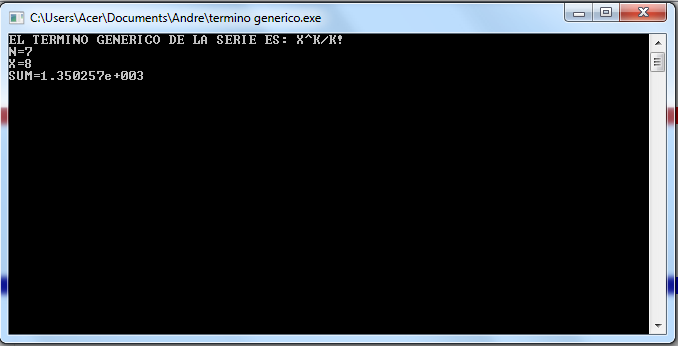


Programa depurado

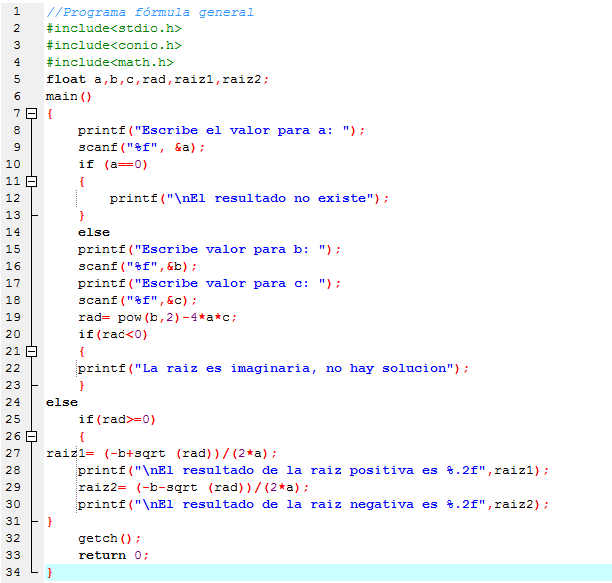


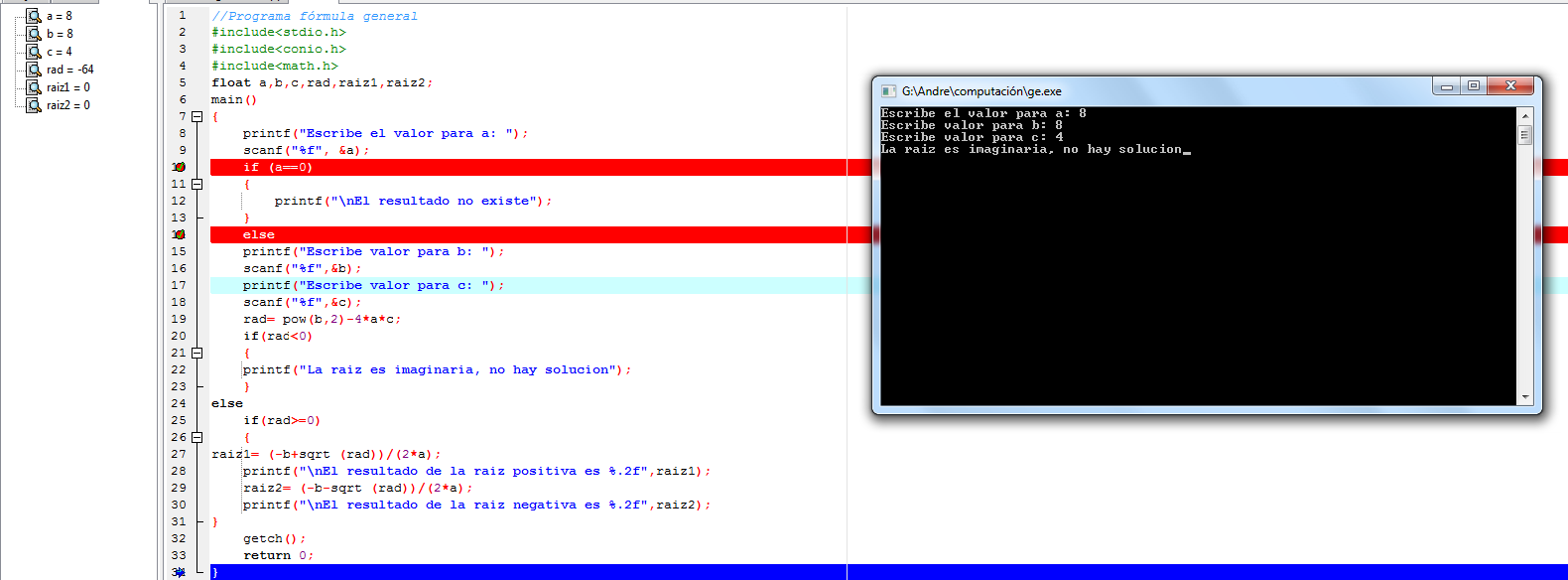


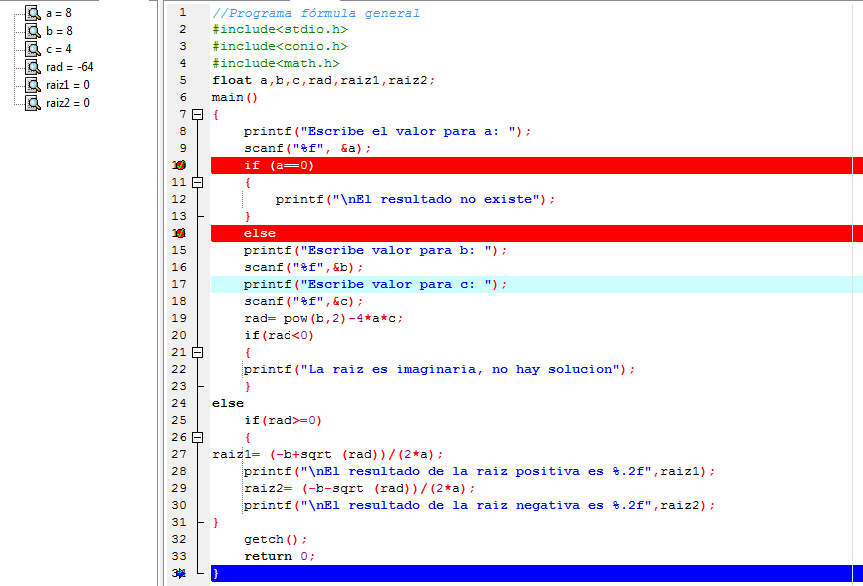


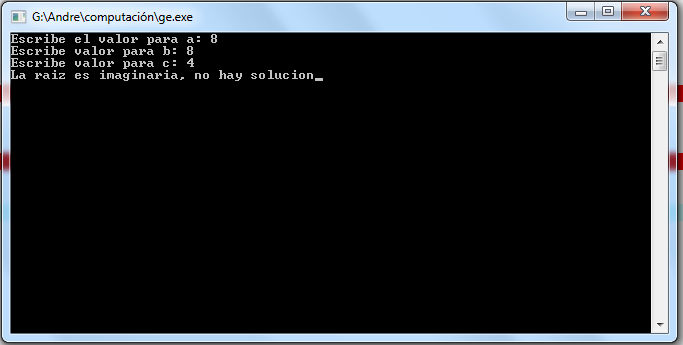


Programa fórmula general









Conclusiones:

Ésta práctica fue importante ya que se pudo aprender la depuración aplicada a un programa la cual es fundamental en el desarrollo del software ya que se permite ver de una nueva manera la ejecución de un programa y el flujo que éste tiene, considero que la depuración es una parte fundamental de la programación ya que es muy útil para detectar las fallas que tiene un programa y que son difícil de observar, además nos permite visualizar que es lo que realmente está ejecutando nuestro proyecto, analizando paso a paso el valor que van tomando las variables en el desarrollo del código ejecutable que a veces es difícil de comprender y la realización de ciertos programas se vuelve ineficiente por lo que el estudio de la depuración es fundamental para la comprensión total de un programa para que de esta manera podamos analizar nuestros errores y lleguemos a ser más eficientes para entender la lógica que tiene la computadora al momento de realizar nuestros respectivos programas.