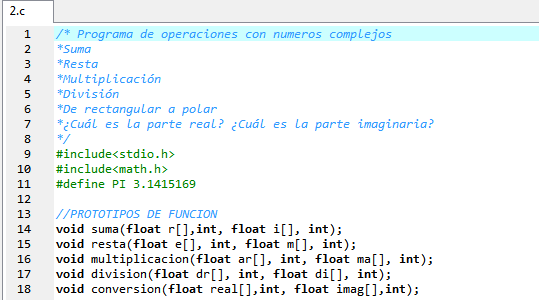
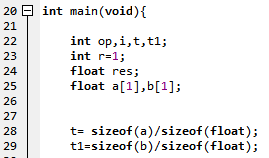
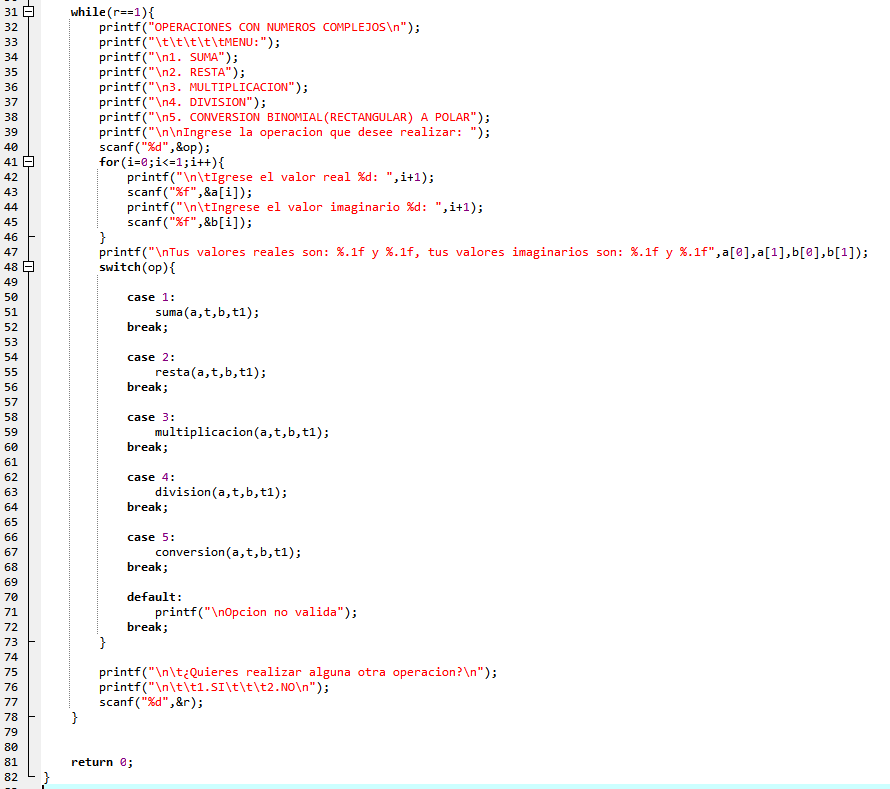
1. Se mandaron a llamar las cabeceras sobre las que se trabajará, también se asignó una constante de tipo PI que se utilizará más adelante para la covnersión de radianes en grados, también se encuentran los prototipos de las funciones que se utilizaron, xomo se pueden leer cada una es para una operación específica, todas fueron de tipo vacia con un parámetro de tipo arreglo (Véase Figura 1).



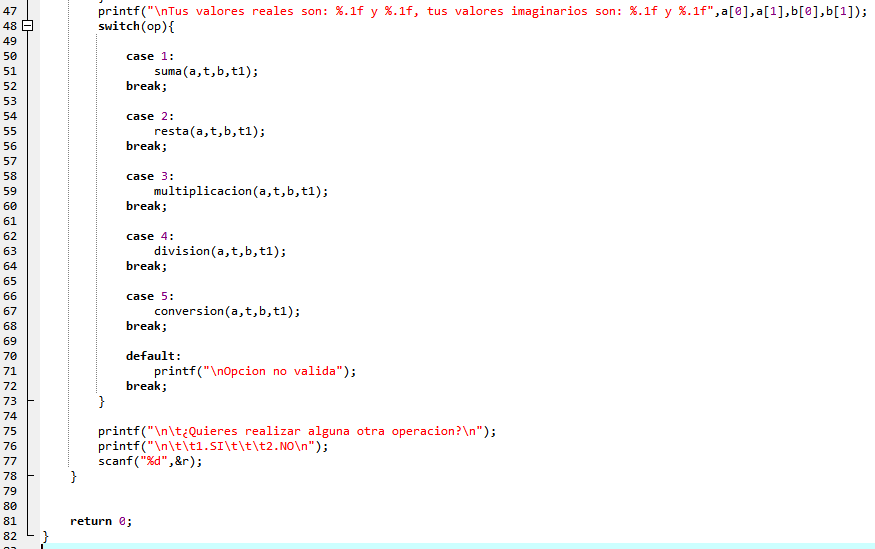
1. Se tiene el programa principal “main”, donde se declararon cinco variables de tipo entero **int,** tres variables de tipo flotante, dos de esas variables flotantes son arreglos destinados a almacenar los valores reales e imaginarios de los datos que ingrese el usuario, se tienen también las asignaciones que tienen en sí la función **sizeof** para determinar el tamaño del arreglo que se indica respecto al tipo de dato que es el arreglo (Véase Figura 2).



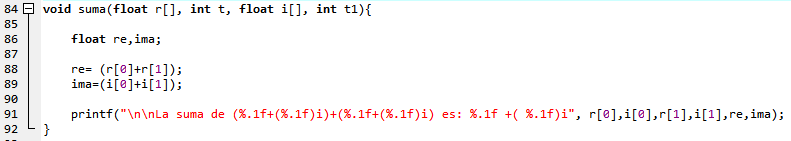
1. Se tiene bucle while con la condición de una respuesta, mientras r sea igual a 1 se va a repetir todo lo que contenga el bucle, se tienen varias impresiones de texto que sirvieron para formar el menú, así como la posibilidad de que ingreses la opción de tu preferencia para que el programa haga esa operación, a su vez te pide que ingreses los datos reales y complejos para la solución de las operaciones que hayas elegido (Véase Figura 3).



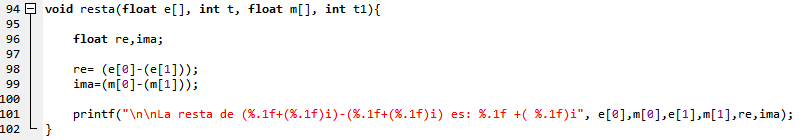
1. Se tienen los diferentes casos a elegir por el usuario por medio de un switch-case, donde si ingresas una opción que no está especificada te va a desplegar un mensaje de opción no válida, a su vez el bucle va a ayudar a que el usuario determine si quiere hacer alguna otra operación. (Véase Figura 4).



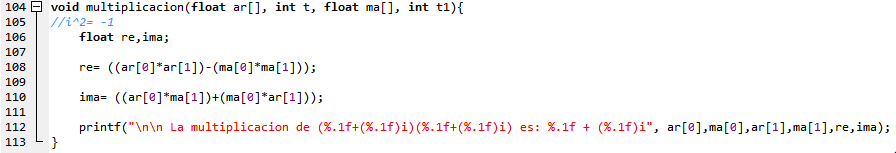
1. Función vacía suma, esta función recibe como parámetros ambos arreglos declarados en el programa principal, así como los tamaños de cada uno que fueron almacenados en variables enteras, en ella se asignó el orden de cómo debía ser realizada la suma de los vectores de acuerdo a la posición del arreglo que se indica en el subíndice del arreglo, a su vez da la opción de imprimir el resultado con el formato que se le asigno haciendo referencia al “i” que se utiliza para hablar de números complejos (Véase Figura 5).



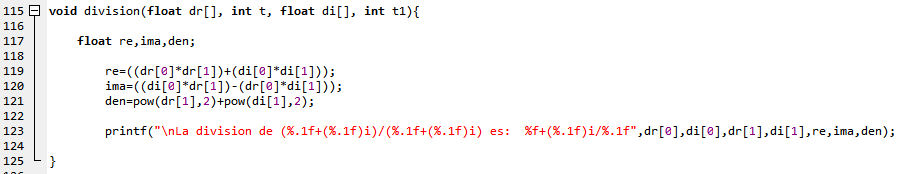
1. En la función vacía resta, se reciben como parámetros los arreglos declarados en el main así como los respectivos tamaños de los mismos, en esta función se realiza la operación resta de números complejos, en la variable re se van a almacenar los resultados de las variables reales y en ima los resultados de la operación entre complejos, también se imprime el resultado de esta operación con su respectivo formato y la referencia de “i” en la parte compleja (Véase Figura 6).



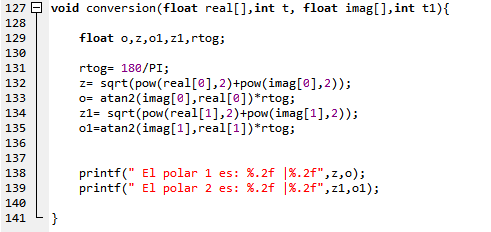
1. En la función vacía llamada multiplicación se reciben como parámetros los arreglos declarados en el main así como los valores enteros de su tamaño, en esta función hay un cambio un poco más notorio respecto a las operaciones que se hacen para determinar la parte real y la imaginaria de esta operación, también se encuentra la impresión del resultado con su debido formato, %.1f se utilizó para limitar la cantidad de decimales en las variables de tipo flotante (Véase Figura 7).



1. Función vacía para hacer divisiones, en este caso se utilizarán tres variables diferentes para almacenar los números reales, imaginarios y el denominador, la función pow se ocupa con la cabecera de math.h, se imprime el resultado con su debido formato (Véase Figura 8).



1. Se hizo una función vacía llamada conversión, esta se va a encargar de realizar la conversión de radianes a grados, en esta parte se van a almacenar distintos resultados para poder llegar a la conversión, ahora se ocupa la constante definida PI, se imprimen resultados con formato para expresar los ángulos (Véase Figura 9).



1. A continuación se tienen las salidas en consola de cada operación ejecutada por el programa.

