

Práctica: Resolución de una Ecuación de Segundo Grado en C++

Objetivo:

El propósito de esta práctica es aprender a resolver una ecuación de segundo grado en C++, utilizando los conocimientos básicos de entrada y salida de datos (cin y cout), declaración de variables y estructuras condicionales. Los alumnos deberán aplicar estos conceptos para calcular el número de soluciones de la ecuación y, en caso de existir, determinar sus valores.

Descripción del Problema:

Una ecuación de segundo grado tiene la forma general:

$$ax^2 + bx + c = 0$$

donde a, b y c son coeficientes reales (decimales). El valor de a no puede ser cero, ya que dejaría de ser una ecuación de segundo grado.

Para resolver la ecuación, se utiliza la siguiente fórmula:

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

donde:

$$x_1 = \frac{-b + \sqrt{b^2 - 4ac}}{2a}$$

$$x_2 = \frac{-b - \sqrt{b^2 - 4ac}}{2a}$$

Instrucciones:

1. Pedir al usuario que introduzca los valores de los coeficientes a, b y c
2. Calcular las soluciones (serán números reales)
3. Mostrarlas por pantalla

Consideraciones:

- Se debe usar adecuadamente la entrada y salida de datos usando cin y cout.
- Se recomienda utilizar la función sqrt() de la librería <cmath> para calcular la raíz cuadrada.

Ejemplo del uso de la función sqrt():

```
#include <cmath>
double raiz = sqrt(25.0); // Calcula la raíz cuadrada de 25, que es 5
cout << "La raíz cuadrada de 25 es: " << raiz << endl;
```

Prueba:

CALCULADORA DE ECUACIONES DE SEGUNDO GRADO

```
#include <cmath>
#include <iostream>
using namespace std;
int main()
{
    double a, b, c;
    cout << "Introduce un valor para a: ";
    cin >> a;
    cout << "Introduce un valor para b: ";
    cin >> b;
    cout << "Introduce un valor para c: ";
    cin >> c;
    double equaci = (b * b) - 4 * a * c;
    if (equaci < 0) {
        cout << "No hay soluciones reales." << endl;
    }
    else {
        double raiz = sqrt(equaci);
        double resultado1 = (-b + raiz) / (2 * a);
        double resultado2 = (-b - raiz) / (2 * a);
        cout << "Las soluciones son: " << endl;
        cout << "X = " << resultado1 << endl;
        cout << "Y = " << resultado2 << endl;
    }
    return 0;
}
```

Introduce el coeficiente a: 1
Introduce el coeficiente b: -3
Introduce el coeficiente c: 2

Preguntas:

1. ¿Qué resultado has obtenido?

He obtenido el resultado de 2 y 1

```
Introduce un valor para a: 1
Introduce un valor para b: -3
Introduce un valor para c: 2
Las soluciones son:
X = 2
Y = 1
```

Daniel Garcia Brun

2. ¿Es correcto el resultado obtenido?

Si es correcto

3. ¿Cómo validas si las soluciones obtenidas son correctas?

El resultado obtenido por el programa lo he pasado por una calculadora online (Calculadora online)

4. ¿Hemos probado todos los casos posibles?

Del problema proporcionado si, aunque para verificarlo debería probar con muchos mas valores y valores diversos

5. Prueba ahora con los siguientes valores:

a.)

Introduce el coeficiente a: 1
Introduce el coeficiente b: 2
Introduce el coeficiente c: 5

```
Introduce un valor para a: 1
Introduce un valor para b: 2
Introduce un valor para c: 5
No hay soluciones reales.
```

Daniel Garcia Brun

b)

Introduce el coeficiente a: 1
Introduce el coeficiente b: 2
Introduce el coeficiente c: 1

```
Introduce un valor para a: 1
Introduce un valor para b: 2
Introduce un valor para c: 1
Las soluciones son:
X = -1
Y = -1
```

Daniel Garcia Brun

6. ¿Tiene solución? En caso afirmativo di cuántas.

Si, solo tiene solución el problema b el cual al ser una solución de una ecuación de segundo grado nos da dos resultados el caso en el que la ecuación es:





$$-b + (\text{raíz} = (b*b) - 4 * a * c)$$

Y la opuesta que es:

$$-b - (\text{raíz} = (b*b) - 4 * a * c)$$

7. Depura el código poniendo breakpoints y examinando el contenido de las variables para descubrir qué línea de código de problemas. Haz capturas de pantalla con ShareX

Como se puede ver el error se produce debido a que la raíz da un resultado negativo

Nombre	Valor	Tipo
 a	1.0000000000000000	double
 b	2.0000000000000000	double
 c	5.0000000000000000	double
 equaci	-9.2559631349317831e+61	double

Daniel García Brun

8. Modifica tu código para que funcione bien en todos los casos. Añade comentarios en el código donde expliques lo que has hecho

```
#include <cmath>
#include <iostream>
using namespace std;

int main()
{
    double a, b, c;

    // Solicita al usuario que introduzca los valores de a, b y c
    cout << "Introduce un valor para a: ";
    cin >> a;

    cout << "Introduce un valor para b: ";
    cin >> b;

    cout << "Introduce un valor para c: ";
    cin >> c;

    // Calcula el discriminante de la ecuación cuadrada
    double discriminante = (b * b) - 4 * a * c;

    // Verifica si el discriminante es negativo, cero o positivo
    if (discriminante < 0) {
        // Si el discriminante es negativo, no hay soluciones reales
        cout << "No hay soluciones reales." << endl;
    }
    else if (discriminante == 0) {
        // Si el discriminante es cero, hay una solución real
        double resultado = -b / (2 * a);
        cout << "La solución es: " << endl;
        cout << "X = " << resultado << endl;
    }
}
```

```

else if (b == 0) {
    //Si a es cero no puede haber soluciones debido a que no se puede
elevant
    cout << "b no puede ser 0, porfavor introduce otro valor";
}
else {
    // Si el discriminante es positivo, hay dos soluciones reales
    double raiz = sqrt(discriminante);
    double resultado1 = (-b + raiz) / (2 * a);
    double resultado2 = (-b - raiz) / (2 * a);

    cout << "Las soluciones son: " << endl;
    cout << "X1 = " << resultado1 << endl;
    cout << "X2 = " << resultado2 << endl;
}

return 0;
}

```

Como se puede ver el código lo he arreglado añadiendo que si X1 y X2 tienen el mismo resultado imprima solo una vez, si la raíz es negativa no tiene solución, si b es cero debido a que 0 no se puede elevar y por último el caso más común en el que tenemos dos resultados.