ECE - 270: Computer Methods in ECE

Assignment #2 - Quadratic Equation Solver

Taylor Tomblin

September 27, 2022

1 Statement of the Problem

This document will describe how to write a C program which allows the user to enter the coefficients of a quadratic equation and solve for it's roots. The results will then print to the screen. The program will handle 3 possible cases: distinct roots, repeated roots, and complex roots.

2 Description of Solution

2.1 Analysis

The quadratic equation has two general solutions:

$$x_1 = \frac{-b + \sqrt{b^2 - 4ac}}{2a},\ x_2 = \frac{-b - \sqrt{b^2 - 4ac}}{2a}$$

The discriminant is used to determine the nature of the roots of a quadratic equation. The discriminant follows:

$$d = b^2 - 4ac$$

The discriminant can be separated into 3 different cases: repeated roots, real distinct roots, and complex roots.

Case 1: Real Distinct Roots. If $d > 0$, then the quadratic equation has two solutions:

$$x_1 = \frac{-b+\sqrt{d}}{2a} \text{ and } x_2 = \frac{-b-\sqrt{d}}{2a}$$

Case 2: Repeated Roots. If $d = 0$, then the quadratic equation has one solution

$$x = \frac{-b}{2a}$$

Case 3: Complex Roots. If $d < 0$, then the quadratic equation has no read solution.

2.2 Code Implementation

The solution for solving a quadratic equation will first obtain the values of a, b, and c. Once these values have been obtained, the program will then compute the discriminant, using the equation above. Once the discriminant has been calculated, the program will check which case is used and compute the roots for the equation per the case.

3 Testing and Output

Using Visual Studio 2022, a program was created to solve for a quadratic equation. Initially approaching the problem seemed fairly simple, then the problem arose of figuring out how to input each case into the program and having it check to see which case it was. The first thought that came to mind was an "if" statement. Knowing that the equation could only have 1 of 3 distinct cases, that changed the initial "if" statement to an "if-else" statement.

Once the code was complete, it was time to test the code. Initially, testing the code gave a "-nan" in front of the i value. This was due to a typo, which was quickly corrected and produced an expected result. Attempting each case that could present in a quadratic equation proved as follows:

Case 1:

```
Please enter the value of a: 1
Please enter the value of b: -5
Please enter the value of c: 6


Your equation is : 1.000000 x^2 + -5.000000 x + 6.000000

Presenting Case 1: The answers to this quadratic equation have two distinct
   roots.

x1 = 3.00 and x2 = 2.00
```

Case 2:

```
Please enter the value of a: 1
Please enter the value of b: 4
Please enter the value of c: 4
```

```
Your equation is : 1.000000 x^2 + 4.000000 x + 4.000000

Presenting Case 2: The answers to this quadratic equation have the same root.

x1 = x2 = -2.00
```

Case 3:

```
Please enter the value of a: 7

Please enter the value of b: 2

Please enter the value of c: 9


Your equation is : 7.000000 x^2 + 2.000000 x + 9.000000

Presenting Case 3: The answers to this quadratic equation are complex.

x1 = -0.14 + 1.12i and x2 = -0.14 - 1.12i
```

---

4 Code

```c
#define _CRT_SECURE_NO_WARNINGS

#include <stdio.h>
#include <stdlib.h>
#include <math.h>
#include <time.h>

int main() {
   float a, b, c, d;
   float x1, x2;
   float real, fake;

   //Enter values of a, b, c
   printf("Please enter the value of a: ");
   scanf("%f", &a);

   printf("Please enter the value of b: ");
   scanf("%f", &b);

   printf("Please enter the value of c: ");
   scanf("%f", &c);

   //Print the equation
   printf("\n\nYour equation is : %f x^2 + %f x + %f", a, b, c);

   //Math
   d = b * b - 4 * a * c;

   //Find roots for equation
   if (d > 0) {
      //Real and different
      printf("\n\nPresenting Case 1: The answers to this quadratic equation have
          two distinct roots.");

      x1 = (-b + sqrt(d)) / (2 * a);
      x2 = (-b - sqrt(d)) / (2 * a);

      printf("\n\nx1 = %.2lf and x2 = %.2lf\n\n", x1, x2);
   } else if (d == 0) {
      //Real and same
      printf("\n\nPresenting Case 2: The answers to this quadratic equation have
          the same root.");
```

```
      x1 = x2 = -b / (2 * a);

      printf("\nx1 = x2 = %.2f\n\n", x1, x2);
   }
   else {
      //Imaginary (Complex: Case 3)
      printf("\n\nPresenting Case 3: The answers to this quadratic equation are
          complex.");

      real = -b / (2 * a);
      fake = sqrt(-d) / (2 * a);

      printf("\nx1 = %.2f + %.2fi and x2 = %.2f - %.2fi\n\n", real, fake, real,
          fake);
   }
}
```