

# **Project One**

Due @ 11:55 PM on Wednesday 9/17/2015

## **Objective:**

1. Become familiar with looping and control structures
2. Become familiar with functions and call by value vs. call by reference.
3. Become familiar with arithmetic in C++

## **Description:**

Create a program that provides the solution to a quadratic equation.

## **Background:**

A quadratic equation can be generalized by equation below:

$$f(x)=ax^2+bx+c$$

Closed form solutions can be found for the zeros of a quadratic function conveniently. That is the locations (x) where the function is equal to zero can be found by the following equation:

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

Note that depending on the sign of the expression under the square root, your system (problem) may have complex solutions. Therefore your program should be able to produce real number, or complex number solutions.

## **Your program:**

- Should prompt the user to choose between quitting, or solving another problem. ASCII character “q” should denote quitting of the session, and “s” should denote solving another problem
- If “q” is depressed, the program to terminate execution
- If “s” is depressed, the program should proceed to obtain the three values a, b, and c that are needed to define a quadratic equation. After this step, your program should produce the two correct solutions.
- If your program receives any other input, it should display the choices until one of “q” or “s” is produced
- Since it is possible to have complex numbers as your solutions, your program should be prepared to calculate and display the complex numbers

## **Program requirements:**

- Your program should be properly commented
- Your program should require no math or additional libraries and should only declare primitive available types
- `int main()`

**Purpose:** Acts as the entry point of the program. Manages user interaction, program flow, and repeatedly solves quadratic equations until the user chooses to quit.

**Inputs:** User keystrokes ( ' s ' for solve, ' q ' for quit, any other input prompts the menu again).

**Outputs:** Controls overall program execution, calls other functions to perform tasks, and prints results to the console.

**Details:** Displays a menu prompting the user to choose an option ( ' s ' = solve, ' q ' = quit).

1. If the user chooses ' q ', the program terminates.
  2. If the user chooses ' s ', the program will display the real solutions, or display “Equation does not have real solutions” and continue.
  3. If the user enters invalid input, re-displays the menu until valid input is given.
  4. Loops back to the menu until ' q ' is chosen.
- **void SolveQuadratic(a, b, c, &x1 &x2)**

**Purpose:** Computes the roots of the quadratic equation  $ax^2+bx+c=0$ .

**Inputs:** a, b, c → coefficients of the quadratic equation (passed by value).

**Outputs:** x1, x2 → the two solutions (passed by reference). These may be equal (double root), distinct real numbers, or complex numbers (depending on the discriminant).

**Details:** Uses the quadratic formula to calculate solutions and assigns them to x1 and x2.
  - **void PrintSolution(x1, x2)**

**Purpose:** Displays the solutions of the quadratic equation to the user in a readable format.

**Inputs:** x1, x2 → the two computed solutions (could be real or complex).

**Outputs:** Prints the solutions to standard output (console).

**Details:** Handles the formatting of results in a comprehensible form such as  $x1 = 5$ ,  $x2 = -3$ .
  - **void GetConstants(&a, &b, &c)**

**Purpose:** Collects user input for the quadratic coefficients.

**Inputs:** References to a, b, c (so they can be updated with user-provided values).

**Outputs:** Stores the values entered by the user into a, b, and c.

**Details:** Ensures that values are valid numeric inputs before returning them to the caller.
  - **bool CheckSolution(a, b, c)**

**Purpose:** Verifies that the solutions computed for the quadratic equation are correct.

**Inputs:** a, b, c → coefficients of the quadratic equation.

**Outputs:** Returns `true` if the computed solutions satisfy the equation  $ax^2+bx+c=0$  (within floating-point tolerance), otherwise returns `false`.

**Details:** Evaluates the quadratic function at each solution and checks if the result is approximately zero.

## Some examples :

$a=1, b=2, c=1, x1=-1, x2=-1$