# **Technical specifications for software** «Development of a system for finding real roots of a quadratic equation»

**Done By Rami Alshalabi   
SE (eng) 1-21**

# Quadratic Equation Solver Requirements

## Goal

To develop a reliable and user-friendly program that accurately solves quadratic equations of the form ax² + bx + c = 0, providing real roots when they exist and appropriate information when they don't.

## Functional Requirements

1. Input Handling:
   * FR1.1: The program shall accept three numerical inputs: a, b, and c, representing the coefficients of the quadratic equation ax² + bx + c = 0.
   * FR1.2: The program shall allow input of integer and floating-point numbers.
   * FR1.3: The program shall handle zero as a valid input for any coefficient.
2. Equation Solving:
   * FR2.1: The program shall calculate the discriminant (b² - 4ac) to determine the nature of the roots.
   * FR2.2: The program shall compute two distinct real roots when the discriminant is positive.
   * FR2.3: The program shall compute one real root when the discriminant is zero.
   * FR2.4: The program shall indicate when no real roots exist (when the discriminant is negative).
   * FR2.5: The program shall handle special cases:
     + FR2.5.1: When a = 0, solve as a linear equation if b ≠ 0.
     + FR2.5.2: When a = 0 and b = 0, indicate if the equation has no solution or infinite solutions.
3. Output:
   * FR3.1: The program shall display the calculated roots with a precision of four decimal places.
   * FR3.2: The program shall provide clear messaging for special cases (e.g., "No real roots", "Infinite solutions").
4. User Interface: // nfr
   * FR4.1: The program shall provide a clear interface for users to input coefficients.
   * FR4.2: The program shall include a "Solve" or similar action to initiate the calculation.
   * FR4.3: The program shall display the results prominently after calculation.

## Non-Functional Requirements

1. Performance:
   * NFR1.1: The program shall provide results within 100 milliseconds of the user initiating the solve action.
   * NFR1.2: The program shall handle coefficient values ranging from -1e10 to 1e10.
2. Usability:
   * NFR2.1: The user interface shall be intuitive, requiring no specialized mathematical knowledge beyond understanding of quadratic equations.
   * NFR2.2: Error messages shall be clear and guide the user towards correct input.
3. Reliability:
   * NFR3.1: The program shall produce accurate results for all valid inputs within the specified range.
   * NFR3.2: The program shall not crash or freeze when given invalid inputs.
4. Maintainability:
   * NFR4.1: The code shall be well-documented with inline comments explaining complex calculations.
   * NFR4.2: The program shall be modular, separating the UI logic from the equation-solving logic.
5. Testability:
   * NFR5.1: The core equation-solving function shall be easily testable in isolation from the UI.
   * NFR5.2: The program shall include a comprehensive set of unit tests covering all identified test cases.
6. Compatibility:
   * NFR6.1: The program shall function correctly on major web browsers (Chrome, Firefox, Safari, Edge).
   * NFR6.2: The program shall be responsive and usable on both desktop and mobile devices.
7. Accessibility:
   * NFR7.1: The program shall be navigable using keyboard inputs.
   * NFR7.2: The program shall use sufficient color contrast for readability.

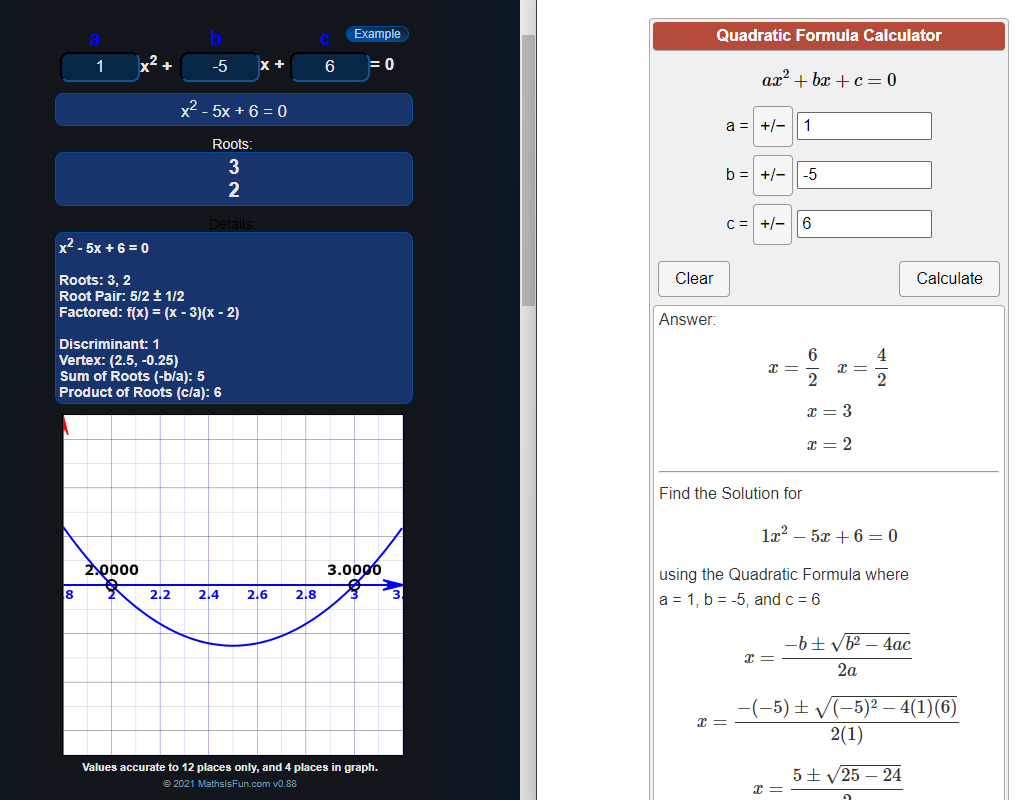
# Sources of development:

* <https://www.calculatorsoup.com/calculators/algebra/quadratic-formula-calculator.php>
* <https://www.mathsisfun.com/quadratic-equation-solver.html>

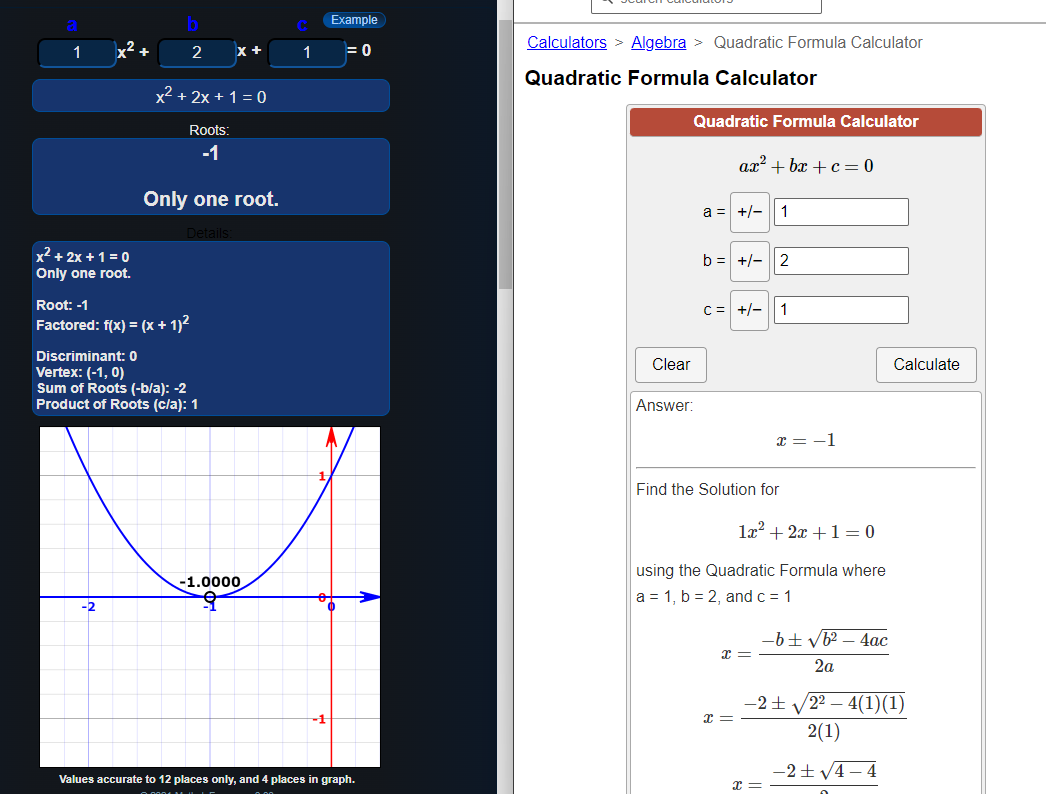
| **Test** | **a** | **b** | **c** | **Expected result** | **What is checked?** |
| --- | --- | --- | --- | --- | --- |
| 1 | 1 | -5 | 6 | x1 = 3, x2 = 2 | Two distinct real roots |
| 2 | 1 | 2 | 1 | x = -1 | One real root (discriminant = 0) |
| 3 | 1 | 0 | 1 | No real roots | Complex roots |
| 4 | 0 | 0 | 1 | No solution (A shouldn’t be 0) | Invalid equation (a = 0, b = 0) |
| 5 | 1 | 0 | 0 | x = 0 | One root at x = 0 |
| 6 | -1 | 0 | 4 | x1 = 2, x2 = -2 | Negative 'a' coefficient /// |
| 7 | 0.5 | 1 | 0.5 | x = -1 | Decimal coefficients |
| 8 | 100 | 2000 | 10000 | x = -10 | Large coefficients |
| 9 | 1e-10 | 1 | 1 | x1 ≈ -1.000, x2 ≈ -9999999999 | Very small 'a' coefficient |
| 10 | A | B | C | Error | Check Symbols. |

1. Tests:

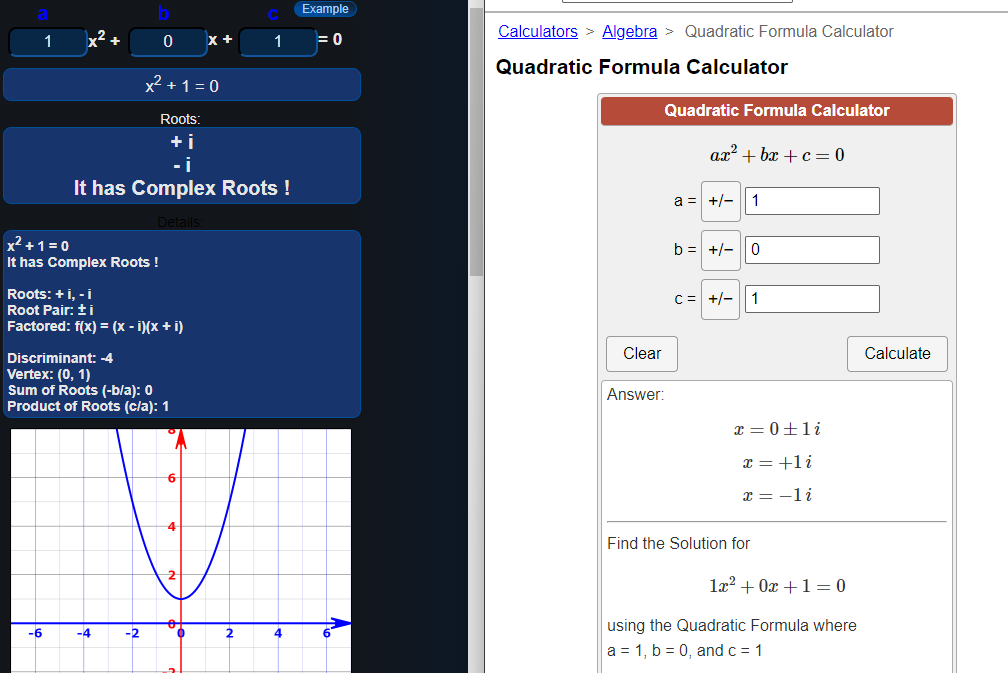
3.1



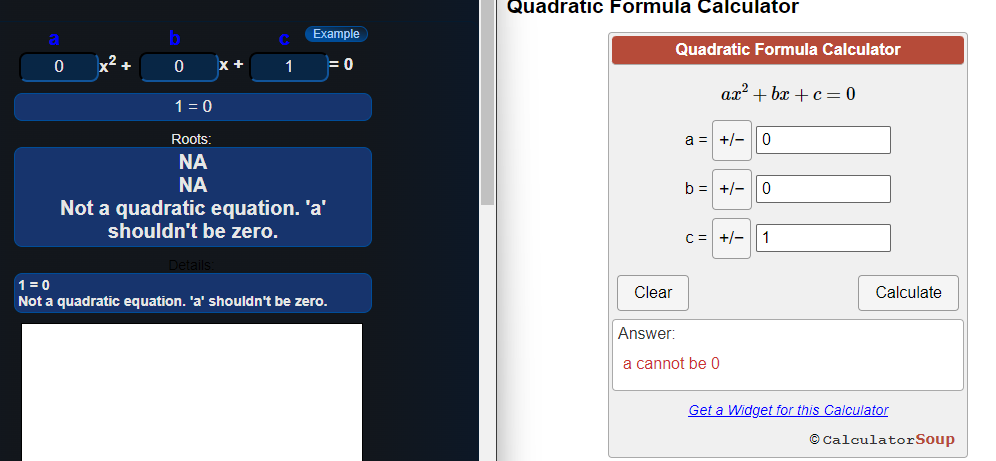
3.2



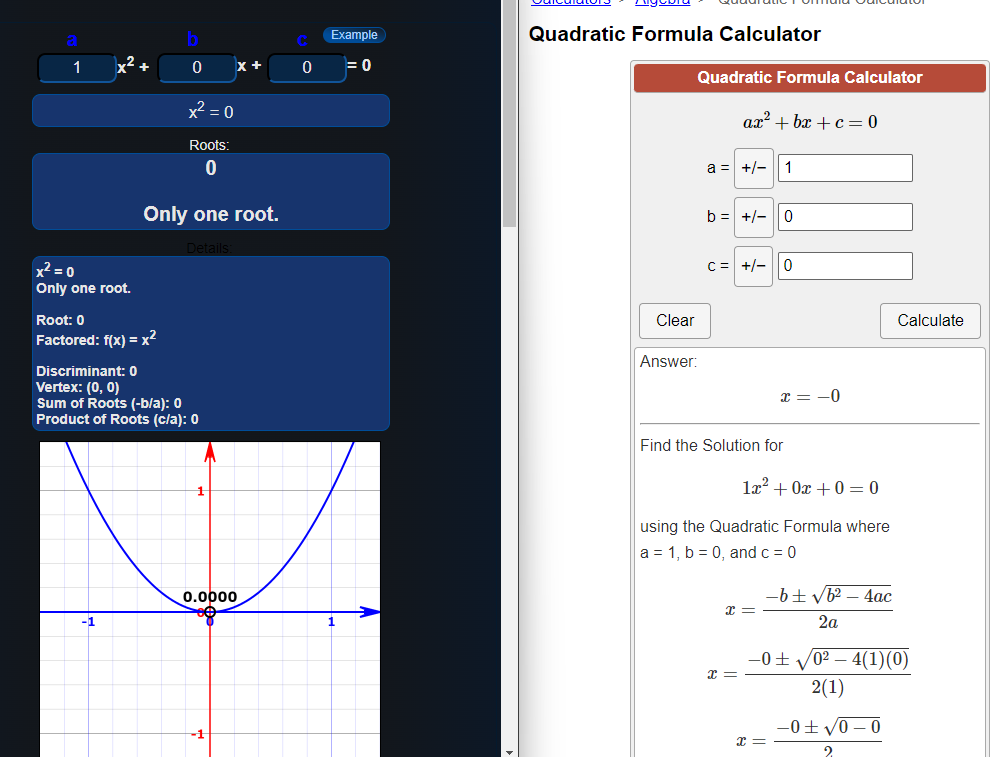
3.3



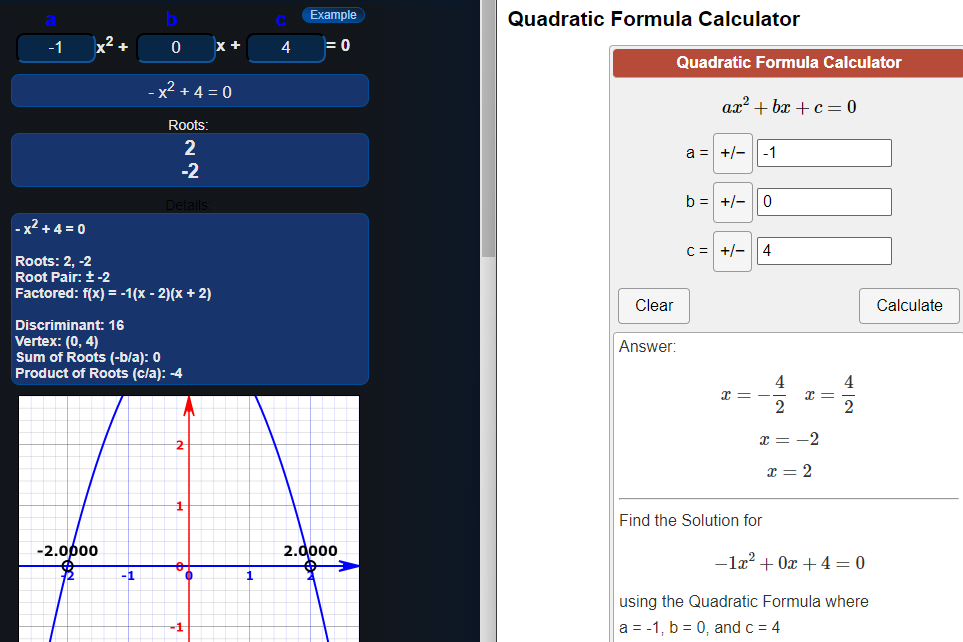
3.4



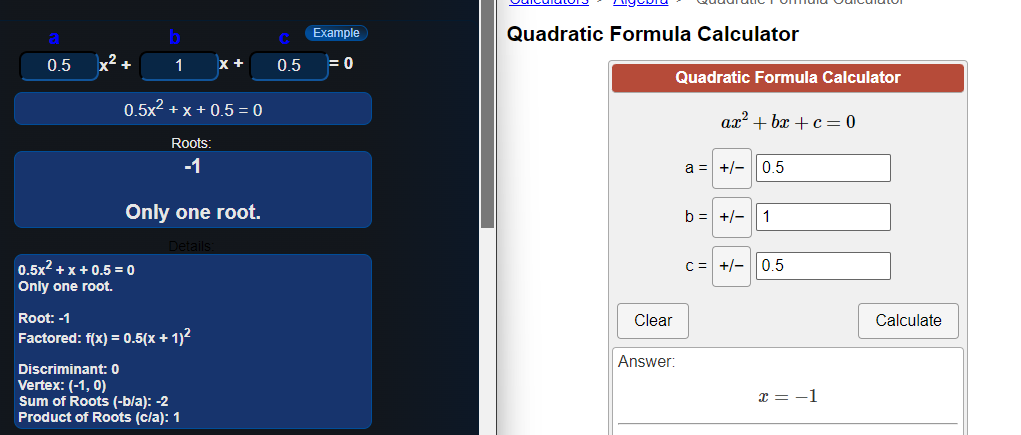
3.5



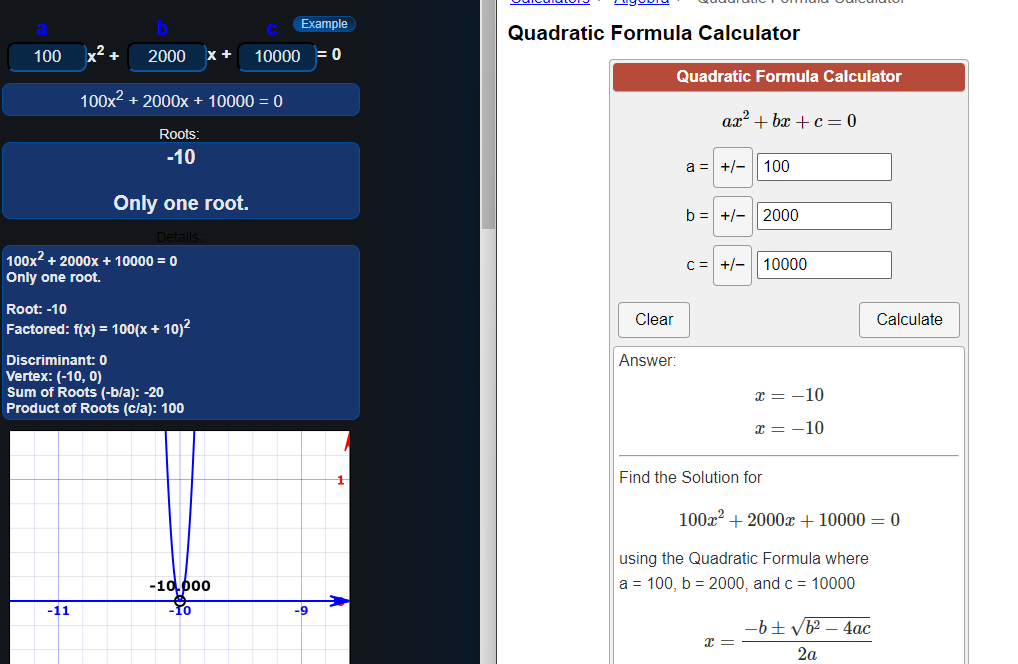
3.6



3.7



3.8



3.9

