Assignment 1

Brett Bonine
ASTR 5900: Numerical Methods

February 10, 2021

1 Problem 1

Write a code in whatever compiled language you prefer to solve a quadratic equation. Test it on a known problem and show that it works.

Answer: The standard form the the quadratic equation is given by

$$ax^2 + bx + c = 0\,. \tag{1}$$

The program 'quad_1.cpp' takes as raw input each of the coefficients a,b, and c and then returns the two roots. As a test, feel free to check the quadratic equation

$$x^2 + 5x + 6 = 0\,. \tag{2}$$

where a = 1, b = 5, and c =6. The expected roots of this equation are x = -2 and x = -3.

2 Problem 2

Make the solver a function or subroutine in a separate file.

Answer: The code for both problems 2 and 3 are contained in the directory 'src/prob_2'. For this problem, the solver was moved to a separate file called "function_2.cpp". A new "quad_2.cpp" was created to call the solver and allow for raw input.

3 Problem 3

Write a makefile that compiles and links your main code and solver

Answer: The makefile can be run in the terminal with the command 'make'. This links the three files 'quad_3.cpp', 'function_3.cpp', and 'quad_3.h'. Doing this produces an executable which can be run with './quad_3'.

4 Problem 4

4.1 Part a

Try some values that give imaginary roots and describe your results. How can the code be changed to handle these?

Answer: The solver for problem 1 is not equipped to handle imaginary roots. If you attempt to solve the quadratic equation

$$x^2 + 4x + 5 = 0\,. \tag{3}$$

using this solver, the code will output 'nan' as the value for both roots. A bandaid solution to this problem was implemented in the solver discussed in problems 2 and 3. In order to check for imaginary roots, the program 'main.cpp' now checks to see if condition

$$4ac > b\,. \tag{4}$$

after the user inputs the values for a,b, and c. If this condition is met, then we know that the program containing the functions will attempt to take the square root of a negative number and return nans. To avoid this, the program now terminates and informs the user that the chosen quadratic equation will return imaginary roots.

4.2 Part b

Try the equation

$$x^2 - 200000x + 1 = 0\,. \tag{5}$$

using both the standard method and the method discussed in class. Discuss results and any advantages or disadvantages.
Answer: In class, we discussed how the product of the roots of a quadratic equation should give c/a

(for rational roots). For this particular equation $a = 1$, so we can construct the second root from the first simply by

$$root_2 = \frac{c}{root_1}\,. \tag{6}$$

Included in the function_3.cpp file is a new function that divides the coefficient c by the output of the root1 function. The output is shown below:

```
----------------------------------------------------
 Version 3: Now finding the second root from the first!
 Please enter the coefficients of your choice.
----------------------------------------------------
Enter a: 1
Enter b: -200000
Enter c: 1

Standard method:
----------------------------------------------------
The first root is: 199999.999995
The second root is: 4.999994416721165e-06

Method from class: Divide previous root by c
----------------------------------------------------
The second root can be found from the first as: 5.000000000125e-06
```

Evidently the main advantage of this new method is its precision. The relationship between the two roots is analytic; if you have the first root, a simple calculation can be preformed to find the second. Using the first method involves evaluating two separate functions, ultimately sacrificing precision.

In lecture we were warned about the computation time required for division, so it is likely that the accuracy of the new method comes at a cost of run time (though any increase in run-time probably isn't significant for a program as small as this.)