Homework Program #1  
Bisection and Newton’s Iteration

The purpose of this project is to see how using different algorithms to solve for the roots of a function affects the runtime and convergence of the x and y values.

Write a program to solve x2 – 56 = 0.

Your program should be able to find both roots (positive and negative) by using different “seed” values (for bisection, *a* and *b*; for Newton’s , x0).

1. For Bisection Method, (pg 49, Algorithm 2.1)  
   1) try a = 0, b = 10  
   2) try a = 7, b = 8  
   3) try a = 0, b = -10  
   4) try a = -8, b = -7
2. For Newton’s Method, (pg. 67, Algorithm 2.3)  
   1) try x0 = 7  
   2) try x0 = -7  
   3) try x0 = 0

For each part, calculate the answers using a while loop with a counter, for both FLOAT and DOUBLE values and print out the iterated values. Calculate the answers for an “appropriate” value for x   
 (example: while b-a > TOL, with TOL = 10-3)   
but remember that the value for float-type will be different from the value for double-type; then compare the value for f(x) (should be “close to” 0)

What do you notice about the counters for the different data types, “seed” values, and methods?

BONUS:   
Print a table of the computations with *x* in **binary** to illustrate the actual convergence in terms of the bits.

DUE: Tuesday September 28, at 12:15 PM.