Homework 8 & 9 – Reference Parameter Functions

This homework is intended to be a follow-on to the reference parameter InClass assignment. Like that assignment, you write some functions that use reference parameters. It should help you better understand (or confirm your understanding) how the 2 types of function parameters work, value parameters and reference parameters. By default, **value parameters** are used to pass information **only into** a function by assigning a value from an *argument* (also called *actual parameters*) in the **calling function** to the *formal parameter* in the **called function** definition. **Reference parameters**, on the other hand, can be used to pass information **also out of** a called function by assigning a value (inside the function) to a reference formal parameter variable.

Like for Homework3 & 4, you will also submit this assignment separately to Homework**9 for hand grading by the TAs** based on its formatting, style, and documentation. A grading rubric will be published on Piazza. Note that you must have file header documentation as well as header documentation for every function -- see the rock-paper-scissors example in the Admin Module on Canvas for an example!

Program Features

You must write these functions:

- `bool gotRoots(double a, double b, double c, double &r1, double &r2);`
  - r1 and r2 are the x values when y=0 in a quadratic, e.g. $y = 3x^2 + 12x - 36$ so that a=3, b=12, and c=-36
  - Modify the code from the class notes. For students who are uncomfortable with math: you do not need to understand the math to do this; just implement the following 2 bullet points.
  - return false, when (temp < 0) in the expression **temp = b * b - 4.0 * a * c**
  - return true, when temp > 0, and set r1 and r2 according to the class notes
  - See http://www.purplemath.com/modules/quadform2.htm if you want to understand the math details
- `int openAndReadNums(string filename, ifstream& fn, double&n1,double &n2, double &n3);`
  - Modify the InClass9 code for this function to open a file with filename, using fn, and read 3 double numbers
  - Add a third reference parameter number to return from the function/file
  - Add **`if (filename == "")`** towards the top of the code inside the function. When the parameter is NULL (equal to ""), the file has already been opened in a previous call to the function, therefore skip the open part of this function and just read 3 more numbers from the file. In other words, skip the open() when filename is "" because fn is already set, just read from it.
  - Instead of printing "Error Reading!" when the read of the numbers fails, print "EOF reached!"

Unlike for previous assignments, most of a main() function is not provided below. You must write it using `gotRoots` () and `openAndReadNums()` described above. Note that the Web-CAT test code will run your main() program AND call your functions directly. For example, my test code will execute your main() program AND call `gotRoots(2, 4, 6, x, y)` or `openAndReadNums( "in.txt", file, x, y)`. Your functions must work with your main() to produce output like the sample below, and the functions must also work with any input my test code passes into them and produce the correct values as return values and reference parameter values.

The purpose of the program as a whole is to find the largest root from all of the calls to `gotRoots()`, reading 3 numbers at a time from a file. In more detail, read 3 numbers at a time using `openAndReadNums()`, call `gotRoots()` using the 3 numbers for a, b, and c; and then compare the 2 roots returned to find the largest root of all returned from all the calls to `gotRoots()`. Here is an outline of the code:

```
set largest to -DBL_MAX (This is the negative of the biggest number a double can hold)
prompt & read a filename
call openAndReadNums() using filename
while (  fn )    // while file still reading OK
    if gotRoots()
    set largest = larger of r1 and r2
    call openAndReadNums() using ""
print largest root
```

Sample Input & Output (Create your own input files so you know what the output should be.)

```
Enter a filename to read from: in2.in
EOF reached!
Largest root: 1.39
```

If the file does not exist, print

```
Error opening file!
```

Submit Your Work to Web-CAT

The assignment will be up on Web-CAT soon!