

## The Quadratic Formula

**Starting with this lab and all future labs, the lab you turn in will be one file.** Here's what I mean:

Your program should start up and show the user the following for this lab:

Please Select which program you want by typing the number. Type 0 to end the program.

0: End Program

1: Quadratic

2: Pythagorean

Depending on what the user enters (IF statement!!) it goes off to solve the problem.

The next thing that must happen is that this is all **embedded** in a loop so that I only invoke your program once and can run multiple things. Here's an easy way to do this:

```
do {  
    // Query user as to what program to run ... see above  
    programNum = in.nextInt();  
    if (programNum == 1) etc.  
} while (programNum !=0);  
  
System.out.println("Goodbye");
```

With this as the outermost statement in all the other stuff, it runs over and over until finally the user enters a zero. So, how does this work? In Java, there is a `do { } while(cond);` statement.

It is a **loop**. It enters the "do" and starts executing all the statements (called a **block**) surrounded by `{}`. Then it hits the **while**. If the condition next to the **while** is true then it goes back up and executes the block again. Note: it will execute the block of statements forever

unless you change the condition variable in the loop. In this case we are asking the user for the program number to run. If we get zero, we end the loop.

Now for the problems!!

A common problem in our math classes is to use the quadratic formula to find the roots of a quadratic equation of the form:

$$Ax^2 + Bx + C = 0$$

Here is the quadratic formula again in case you forgot:

$$x = \frac{-B \pm \sqrt{B^2 - 4AC}}{2A}$$

To take the square root of something in Java requires you to call a function called **sqrt**. This function is in the Math library and to use it we need to put the word **Math** in front of the actual square root function. **This will be true of other math functions.** (As a note, we have already been doing this with printing stuff out, e.g. `System.out.println(stuff)`).

This function takes a **double** as an argument and returns a **double** as a result. Here is a thunk of code demonstrating this.

```
double resultvalue;  
double numtotakeroot=42.0;  
  
resultvalue = Math.sqrt(numtotakeroot);
```

**You should search online for other math functions (hint, try “java math class” in google).**

**Your task for this lab is to write a program that calculates the roots of a quadratic formula when the user inputs values for A, B, and C.**

As you might recall, there are three kinds of results you can expect from solving a quadratic equation: two real roots, a double root and two imaginary roots. Your program should be able to:

1. Print out which of these three cases will occur for a given A, B, and C.
2. Print out the actual roots of the equation if the solution is not imaginary.
3. **Challenge:** Print out all roots, including the imaginary roots in standard form (a+bi).

Here are a few equations to test with:

Real Roots:  $x^2 + 2x - 8$

Double Root:  $16x^2 - 40x + 25$

Imaginary Roots:  $2x^2 + 3x + 5$

## The Pythagorean Theorem

Your task is to write a program that solves the Pythagorean Theorem. All the user to input two of the three sides of a right triangle and the program will output the missing side. Here is the formula if you have forgotten:

$$a^2 + b^2 = c^2$$

Some things to consider:

1. We need some way for the user to let the program know which of the variables to solve for. We can do this by having the user enter a specific value that the computer checks for before doing any calculations ... in this case we can use -1 (since a triangle can't have a negative side length). Make sure to let the user know they need to input -1 to indicate the unknown side.
2. Solving for a, b, and c are all going to need their own equations.
3. **Challenge-** Have the program output the measures of the unknown angles in the triangle. (Hint: The math functions **asin()**, **acos()**, and **atan()** will find the inverse of the corresponding trig function.

**As in the prior lab, this program should be constructed to offer a menu to the user to select which "subprogram" to run. Be very careful! When test-driving I will be checking for invalid inputs!**