***CSC 3020***

***Java Programming***

**Lab 06**

**25 points**

**Due 11/05/2020 (10:10 A.M.)**

Assignment Objectives:

■■ To use abstract classes.

■■ To discover polymorphism and dynamic binding.

All labs must be submitted by the Canvas. **No email or hard copy** is accepted. You must follow the following format:

1. Submit your file to the Canvas. You must submit your file on time; otherwise, you will receive zero.
2. You can upload your file as many times as you like. Only the last attempt counts because the last file you uploaded is the only file your instructor will see.
3. There will be several modules on the Canvas. You need to upload your file using the correct module on the Canvas.
4. Name the lab file: *Lab (labt number)*
5. To upload your file(s):

* In Course Navigation, click the ASSIGNMENTS module.
* Click the title of the assignment.
* Click the **Submit** Assignment button.
* Add **File**. ...
* **Submit** Assignment. ...
* View **Submission**.

*It is your responsibility to make sure that the file is uploaded correctly. If you uploaded a wrong file, you receive zero; files will not be accepted after due date even if you have a prove that the file is created before the due date.*

***Make sure you review the Cheating & Plagiarism policy on Canvas.***

**Solution to this assignment will not be posted on Canvas; however, any question can be discussed in the class upon request of a student.**

Download the GeometircObject example from Canvas (Lab 6 link).

* Add a method to class Rectangle that returns the diagonal of a rectangle.

The formula to calculate the diagonal is below



* Write a test program that includes a method with one argument, an array of GeometricObject. The method sums and prints the sum of areas of all the Circle objects, the sum of areas of all the Rectangle objects, and the sum of the diagonals of all the rectangle objects. The method signature is:

**public static void** sumArea(GeometricObject[] a)

* In main method, create an array of four objects (two circles with radius 5 and 6, and two rectangles with length 2 and width 3 for each rectangle) and call **sumArea** method. You must use a loop to process the array.

A sample output is below

