



Université d'Ottawa · University of Ottawa

Faculté de Génie - Faculty of Engineering
ITI1121Z Introduction to Computing II –Assignment 2

Due date: July 3rd , 2023 – 11:30 PM EDT

Professor: **Wassim El Ahmar** (welah096@uottawa.ca)

Important note: Assignments are individual work. We will use software solutions to automate the process of checking similarity between student submissions for further inspection by the professor. Similarity of your submission with any public solution will also be investigated.

Remember the shape class we have been going over since the beginning of the term? The goal of this assignment is to fully implement the **functionality of a shape class**, according to the following:

- You must have an **interface for Shape** *an abstract class can have an interface but cannot be instantiated so no*
- You will develop the following classes implementing Shape:
 - Square
 - Triangle (assume that it is an equilateral triangle, meaning all of its sides are equal)
 - Circle
 - Pentagon
- Each shape must be able to perform the following functionalities:
 - Get the **area** of the shape (10%)
 - Get the **perimeter** of the shape (10%)
 - **Get/set** the **parameters** of a shape: (20%)
 - A circle has a center and a radius
 - A triangle has a center and the length of the side
 - A square has a center and the length of the side
 - A pentagon has a center and the length of the side (all sides of the pentagon are equal)
 - Ability **to move** the shape into a new location according to an offset x and offset y (10%)
- What to submit:
 - Your source code containing the following files **only**:
 - Shape.java
 - Pentagon.java
 - Circle.java
 - Triangle.java
 - Square.java
 - Main.java (creating a minimum of 2 instances for each shape and showing the functionality – 30 %)
 - UML diagram (20%)
- **BONUS:** To make this assignment a bit more interesting, let's have a challenge. I want to know if two shapes are overlapping. That means, for example, if I have a square and a circle, I want to know if they are overlapping (part of the circle will overlap with part of the square). Write a function that accepts two shapes of any type, and returns whether or not those two shapes are overlapping.
 - Rules of the bonus:
 - **Only the first 5 students** to submit a **complete, commented, and fully functional** assignment that includes the bonus function will receive the bonus. Send me an email when you submit the assignment on brightspace with the bonus function so I could take a look at your submission ASAP.
 - I will announce when 3 students have received the bonus. After I make that announcement, please do not send me emails informing me that you have completed the bonus as you will not be eligible for it.

- You are not allowed to use if else statements in the function that is checking for the overlap. (no instance of allowed people....)
- **No questions on the bonus function will be answered – understanding what you should do is part of the challenge!**
- The bonus is 40% extra on this assignment. (3/100 on the final course grade)