DigiPen
INSTITUTE OF TECHNOLOGY

Programming Evaluation

Module 2 – High-level Programming II

Purpose

This assignment will give you some practice with object-oriented programming (classes, objects, constructors, destructors, inheritance, polymorphism, etc.), as well as pointers and dynamic memory allocation.

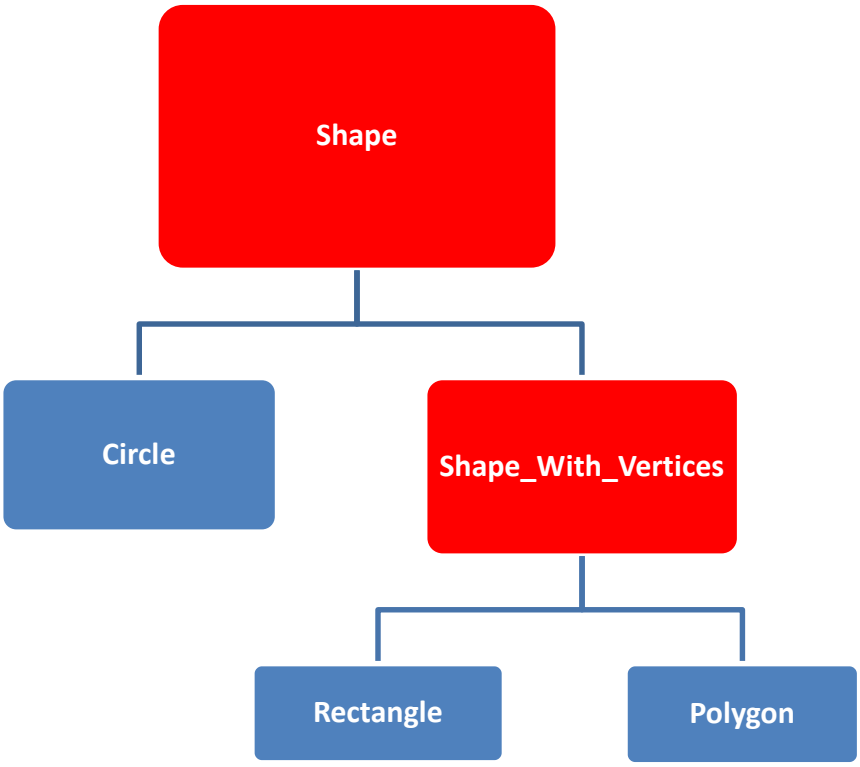
Information

Before writing any code, you should design your full architecture. Draw a diagram containing the hierarchy of all the classes and their content.

You are given a driver ***(main.cpp)*** that is testing the following classes:

- Shape
- Shape_With_Vertices
- Circle
- Rectangle
- Polygon

Here is a diagram that shows you the hierarchy that you are required to follow:



Note: Red nodes represent abstract classes.

Your job is to implement the above classes in order to pass all the tests provided in main.

Note: Some of the tests should lead to a compiler error, so make sure you are getting one when you uncomment that line of code.

Here are **SOME** of the methods that you will find in the above classes. I did not specify if the properties or methods are ***public***, ***protected*** or ***private***. Also, I did not specify if the methods are ***const***, ***virtual*** or ***pure virtual***. It is up to you to decide.

Shape Class	

Methods	`Shape(Point center_)`
	`~Shape()`
	`Point GetCenter()`
	`void SetCenter(float x_, float y_)`
	`void Draw()`

Properties	`Point center;`
Description	Abstract class that serves as the parent of all shapes . In it you will find all properties and methods that belong in all shapes (Center, Draw, etc...)

Shape_With_Vertices Class	

Methods	`Shape_With_Vertices(Point center_, unsigned int number_of_vertices_)`
	`~Shape_With_Vertices()`
	`void SetCenter(float x_, float y_)`

Properties	`Point *vertices;` `unsigned int number_of_vertices;`
Description	Abstract class that serves as the parent of all shapes that contain vertices (Rectangle, Polygon) . In it you will find all properties and methods that belong in all shapes that have vertices. This class will contain a dynamically allocated array of vertices (a vertex is just a point with an x and y position). ***Note: if you move the center of a shape that contains vertices, you need to move the vertices as well.***

Circle Class	
Methods	`Circle(Point center_, unsigned int radius_)`
Properties	`unsigned int radius`

Rectangle Class	
Methods	`Rectangle(Point center_, unsigned int width_, unsigned int height_)`
Properties	`unsigned int width;` `unsigned int height;`

Polygon Class	

Methods	`Polygon(Point center_, unsigned int number_of_vertices_);`
	`Polygon(Point center_, const Point *points_, unsigned int number_of_vertices_);`
	`void SetVertex(unsigned int vertexIdx_, float x_, float y_);`

Once again:

- I did **NOT** provide all the methods that should be found in the above classes. It is up to you to add the remaining ones.
- You are responsible for the architecture ***(public, protected, private, const, virtual, pure virtual, etc…).***
- The tests in **main.cpp** should guide you with the architecture and with the additional methods***.***
- You can add helper functions but they all need to be ***private*** or ***protected***.

Output files are provided so that you can compare your output with the expected output.

Testing your code

A "Grading Scripts" folder is provided in order for you to check if your implementation is correct.

In order to run the tests follow these steps:

- grab the files that contain all of your implementation and place them in the "**Grading Scripts**" folder. The files are:
  - Shape.cpp, Shape.h
  - Shape_With_Vertices.cpp, Shape_With_Vertices.h
  - Circle.cpp, Circle.h
  - Rectangle.cpp , Rectangle.h
  - Polygon.cpp, Polygon.h

- Double click on the **"RunAll.cmd"** file.
  - All tests will run automatically. For every test, you will know if you passed it or failed it. If you failed the test, the scripts will tell you why you failed it.

    NOTE: ***If you are working on a DigiPen lab/classroom machine you need to run the "DigiPen-RunAll.cmd" file instead of "RunAll.cmd"***

What to submit

You must submit the following CPP and Header files:

- Shape.cpp, Shape.h
- Shape_With_Vertices.cpp, Shape_With_Vertices.h
- Circle.cpp, Circle.h
- Rectangle.cpp , Rectangle.h
- Polygon.cpp, Polygon.h

in a single .zip file (go to the class page on Canvas and you will find the assignment submit link).

Do not submit any other files than the ones listed.