

CSC 2111 Lab 12

Objectives:

1. Implement an abstract class
2. Use pointers and references effectively

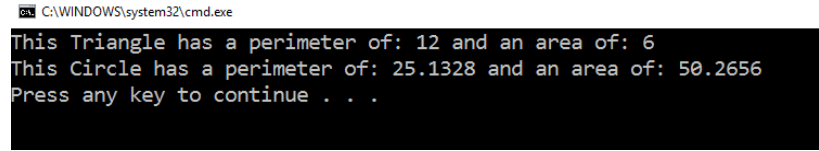
Question 1:

Download **lab12_Q1.cpp**. Triangles, and Circles are Shapes, geometric objects which can have their perimeter and area calculated. Implement a *Shape* abstract class, which can be used in the following manner:

```
void describeShape(Shape &s) {
    double area = s.getArea();
    string type = s.getType();
    double perim = s.getPerimeter();

    cout << "This " << type << " has a perimeter of: "
         << perim << " and an area of: " << area << endl;
}
```

The *Shape* class is provided for you. Implement *Triangle* and *Circle* classes and use the given driver program to produce the following output:



```
C:\WINDOWS\system32\cmd.exe
This Triangle has a perimeter of: 12 and an area of: 6
This Circle has a perimeter of: 25.1328 and an area of: 50.2656
Press any key to continue . . .
```

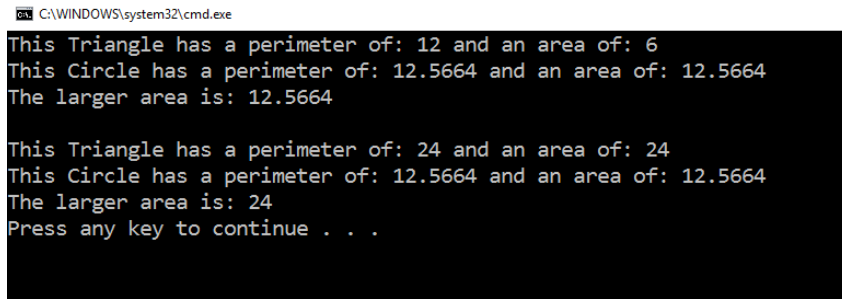
Please do not modify the driver program. No state pertaining to the Triangle or Circle may be stored in the Shape objects. Use Heron's formula for calculating the area of a Triangle: $A = \sqrt{s(s-a)(s-b)(s-c)}$, $s = (a+b+c)/2$

Question 2:

Download **lab12_Q2.cpp**. The following function accepts objects by reference and indicates the object that has larger area by storing a value in the variable pointed to by *result*. Implement the function using the classes defined in **Question 1**.

```
/**  
 * Determines the larger area between two Shape objects  
 * The larger area is stored in result  
 */  
void largerArea(Shape &a, Shape &b, double *result);
```

Use the given driver program to produce the following output:



```
C:\WINDOWS\system32\cmd.exe  
This Triangle has a perimeter of: 12 and an area of: 6  
This Circle has a perimeter of: 12.5664 and an area of: 12.5664  
The larger area is: 12.5664  
  
This Triangle has a perimeter of: 24 and an area of: 24  
This Circle has a perimeter of: 12.5664 and an area of: 12.5664  
The larger area is: 24  
Press any key to continue . . .
```

Note, `largerArea()` must not produce terminal output, the value must be passed to the caller through the result pointer variable.