

Creative Software Programming Practice (week-10-1)

Every assignment will be announced on **Thursday** and should be submitted by next **Tuesday**.

We have practice classes on Wednesdays and Thursdays.

The contents of the practice class are different from the assignments and aim to be completed on the same day.

Assignments will be published on Thursday and must be submitted by the next Tuesday.

In this week **Handed out will be Nov 5, 2020, Due Nov 10, 2020**

Topics

1. Practices

1. Practices

We learned about polymorphism and inheritance until the previous hour.

Therefore, we will practice implementing a program using this. Write a program that works as follows:

1. Define class Shape that has a constructor taking *width* and *height* as parameters (double type).
2. Define class **Circle**, **Triangle** and class **Rectangle** which inherit from class Shape. Each subclass also has a constructor taking *width* and *height* as parameters (double type).

assume that $PI = 3.14$

3. All classes have a member function, `double getArea()`.
4. `Shape::getArea()` is a pure virtual function.
5. Take inputs from the user and create objects of proper type according to the input by new operator, and then put those objects into `std::vector<Shape*> arr`.
6. When the user enters 0, call the `getArea()` of each element of `arr` to print out calculated area. Each element of `arr` must be deallocated after use.
7. Do not use the type casting operator throughout the code.
8. This program should take user input repeatedly

Input:

- *r width, height*
create a rectangle
- *t width, height*
create a triangle
- *c width, height*
create a circle
- 0
print the results and quit the program

Output:

Areas of the shapes

Example:

```
r 10 5
t 2 4
c 3 3
0
50
4
28.26
```

```
// shapes.h
```

```
class Shape {
public:
    double width, height;

    Shape(double width, double height)
    : width(width), height(height) {
    }

    virtual getArea() = 0;
};
```

```
class Rectangle ...
class Triangle ...
class Circle ...
```

```
// shape-main.cc
```

```
#include <iostream>

int main() {

}
```