Programming Two

Tutorial 5: Polymorphism and Virtual Functions

Compiling the Example Code

Compile the example code online, there is quite a bit, and make sure you understand what is going on.

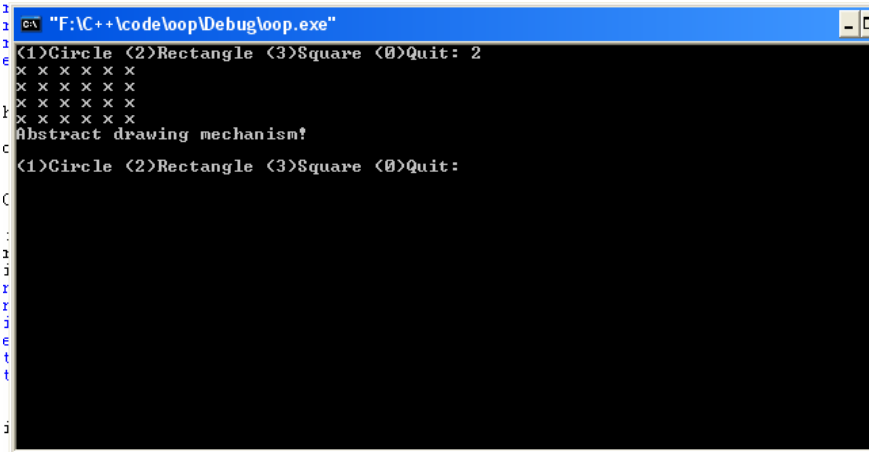
Exercise

Consider the following class hierarchy: *CShape* is the common base class. It has a constructor, with no arguments, and is an abstract class, containing a pure virtual function called *Draw()*. Consider the derived classes *CTriangle*, *CCircle* and *CRectangle*, all of them implementing their own Draw method (mandatory). Write down the corresponding files.

To advance this, use the following as your base class:

```
class Shape
 {
 public:
    Shape(){}
    virtual ~Shape(){}
    virtual long GetArea() = 0;
    virtual long GetPerim()= 0;
    virtual void Draw() = 0;
 private:
 };
```

Define the functions and main so that a similar output to this is produced



In main you will have to use the keyword **new** to define a new instance based on the user's selection and have a pointer to the base class which then points to the new instance of a derived class. Using the pointer then call the derived classes *Draw()* function.

For the circle function you don’t actually have to define a drawing routine, just get it to print a relevant message to the screen.