

THE UNIVERSITY OF MELBOURNE  
School of Computing and Information Systems  
**COMP90041**  
**Programming and Software Development**  
Second Semester, 2018

Fifth Assessed Exercise (lab5)

Submission due Saturday, 20 October 2018, 5:00PM

These exercises are to be assessed, and so **must be done by you alone**. Sophisticated similarity checking software will be used to look for students whose submissions are similar to one another.

1. Write an abstract Java class **Shape** that represents a geometric shape, and create concrete (ie, non-abstract) subclasses **Circle** and **Rectangle**. These classes should all be immutable. All shapes must support the following methods:

**Double perimeter()** Returns the perimeter of the shape. For circles, this is the circumference.

**Double area()** Returns the area of the shape.

**String toString()** Returns a string showing the constructor used to construct the object (that is, it returns a string of the class name, an open parenthesis, the argument(s) to the constructor separated by commas, and a close parenthesis. Do not include any whitespace.

The **Circle** class should have a constructor that takes the radius as input; the **Rectangle** class's constructor should take a width and height as input, in that order. All these arguments should be **Doubles**.

The key requirement is that **Shape** should be a Java type that supports the three methods above. For example, it must be possible to do this:

```
Shape shape = new Circle(1.0);  
System.out.println("Area of unit circle is " + shape.area());
```

Note that the class of the **shape** variable is **Shape**. Declaring it to be of type **Circle** is not good enough.

**Reminder:** The formulas for perimeter and area of these shapes are:

Shape	Perimeter	Area
<b>Circle</b> ( $r$ )	$2\pi r$	$\pi r^2$
<b>Rectangle</b> ( $w, h$ )	$2w + 2h$	$wh$

## Submission

You must submit your project from any one of the student unix servers. Make sure the version of your program source files you wish to submit is on these machines (your files are shared between all of them, so any one will do), then `cd` to the directory holding your source code and issue the command:

```
submit COMP90041 lab5 Shape.java Circle.java Rectangle.java
```

**Important:** you must wait a minute or two (or more if the servers are busy) after submitting, and then issue the command

```
verify COMP90041 lab5 | less
```

This will show you the test results and the marks from your submission, as well as the file(s) you submitted. If the test results show any problems, correct them and submit again. You may submit as often as you like; only your final submission will be assessed.

If you wish to (re-)submit after the project deadline, you may do so by adding “.late” to the end of the project name (*i.e.*, **lab5.late**) in the **submit** and **verify** commands. But note that a penalty, described below, will apply to late submissions, so you should weigh the points you will lose for a late submission against the points you expect to gain by revising your program and submitting again. **It is your responsibility to verify your submission.**

## Late Penalties

Late submissions will incur a penalty of 1% of the possible value of that submission per hour late, including evening and weekend hours. This means that a perfect project that is a little more than 2 days late will lose half the marks. These lab exercises are frequent and of low point value, and your lowest lab mark will be dropped. Except in unusual circumstances, I will not grant extensions for lab submissions.

## Academic Honesty

This lab submission is part of your final assessment, so cheating is not acceptable. Any form of material exchange between students, whether written, electronic or any other medium, is considered cheating, and so is the soliciting of help from electronic newsgroups. Providing undue assistance is considered as serious as receiving it, and in the case of similarities that indicate exchange of more than basic ideas, formal disciplinary action will be taken for all involved parties.