

# CSC72003 Assignment 1

Weight: 20% of your final mark

Due: 19 Apr 2019 10 pm

## Specifications

Your task is to complete various exercises in BlueJ, using the Java language, and to submit these via the MySCU link created for this purpose.

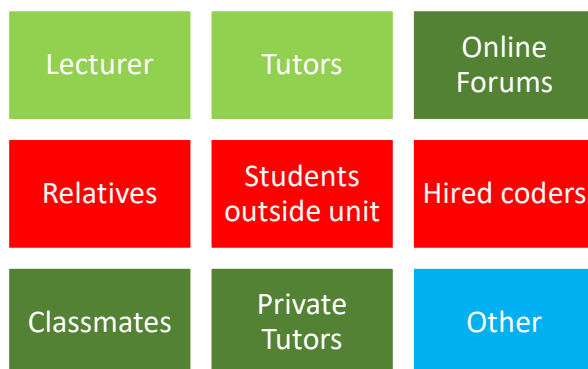
Marking criteria includes:

- Use of **correct coding style**, including the **use of comments**;
- **Accuracy** of coding;
- Use of **suitable coding structures**;
- **Correct submission** and **naming conventions** of assessment items as required.

## Getting Help

This assignment is to be completed individually. It is the opportunity to gain an understanding of the concepts of object oriented programming and coding syntax. It is important that you master these concepts yourself. You are permitted to work from the examples in the study guide or textbook but you must acknowledge assistance from other textbooks or classmates. In particular, you **must not** use online material or help from others, as this would prevent you from mastering these concepts.

**Who can you get help from?** Use this diagram to determine from whom you may seek help with your program.



**Encouraged**

**Attribution Required**

**Ask tutor**

**Not acceptable**

## Assignment Q1

### To be completed in week 1

Create a word document and call your document username-A1Q1.docx. For example, mine would be ahendr10-A1Q1.docx

Open the **house** project from chapter 1 of the book projects.

Open the terminal window and record method calls.

Create a picture with at least six (6) objects (circle, square, triangle, and person), recording all method calls.

Take a screenshot of your picture and add it to the word document you created earlier. Now copy the list of method calls needed to create the picture to the word document after the screenshot.

Add your name and student ID in the footer of the word document, as well as "CSC72003 Ass 1 Q1".

## Assignment Q2

### To be completed in week 2

Create a new BlueJ project called username-A1Q2. For example, mine would be ahendr10-A1Q2.

Click on New Class and give the class a name of **Car**. Make sure "Java" and "class" are selected.

Write a **description** of your new Class in the top comments. Make sure you put in your name as the author and the date as the version you last worked on this exercise.

Add 2 fields that are suitable for a car. 1 field should have a type of **int**, the other should have a type of **String**.

Add a third **boolean** field called **isNew**.

Write a **constructor** for your Car class that takes two (2) parameters - the first of type **int**, the second of type **String**. Set the initial values of the first 2 fields that you created using the parameters.

Write **accessor** methods for all 3 of your fields

Write **mutator** methods for all 3 of your fields

Write a method called **printDetails**, which prints out all the details of a Car object. Please have a look at the 'Printing' section in the Topic 2 study guide to see how to do this.

Take into account the **isNew** status. The printDetails method should print a line saying that the car is new, or the car is not new.

## Assignment Q3

### To be completed in week 3

Create a new BlueJ project called username-A1Q3. For example, mine would be ahendr10-A1Q3.

Create a class, **Assignment**, that contains the following four fields:

- A **String** called **name**
- A **double** called **assignmentMark** (which will store the mark each assignment is worth e.g. for this assignment that you are doing right now the value would be 20)
- A **double** called **studentMark** (stores the mark the student gets in the assignment e.g.15)
- A **String** called **grade**

Make sure you write a description of your new Class in the comments, with your name as author and give the version as the date you last worked on this exercise.

Define a constructor that takes and sets the **name**, **studentMark** and **assignmentMark**.

Also define a **constructor** that takes no parameters and sets the **assignmentMark** to 100.

Create an **accessor** and **mutator** for **studentMark**. The mutator should not let the **studentMark** be set a value greater than the **assignmentMark** (as the student cannot get a mark higher than the assignment is worth) or less than 0. If the user tries to set a value that is not valid a suitable error message should be displayed.

Create a **method** that **calculates** the **grade** for the student. You will need to work out how many percent the student scored in the assignment.

If the student scored:

- Less than 50% the grade will be **fail**
- 50% - 64% the grade will be **pass**
- 65% – 74% the grade will be **credit**
- 75% – 84% the grade will be **distinction**
- Greater than 85% the grade will be **high distinction**

For example, if assignmentMark is 30 and studentMark is 15, the percentage will be 50% so the grade will be set to pass.

Define an **accessor** method to return the value of grade.

## Assignment Q4

### To be completed in week 4

Create a new BlueJ project called your username-A1Q4. For example, mine would be ahendr10-A1Q4.

Create a class called **LotsOfNumbers**, that has one **ArrayList** field called **numbers**, which holds a collection of **Integers**.

Make sure you write a description of your new Class in the comments, with your name as author and give the version as the date you last worked on this exercise.

Define a **constructor** that initialises the ArrayList. Note that you can add any other initialisations that you feel are relevant.

Create methods to **add** elements, **remove** elements and **get** the number of elements in the collection. Make sure you add tests for errors and sensible error messages.

Create a method called **printNumbers**. This method should loop through the collection and print out the elements (each Integer on one line) as determined by the following rules;

- If the Integer is divisible by 3, instead of printing the Integer, it should print **divisible by 3**
- If the Integer is divisible by 5, instead of printing the Integer, it should print **divisible by 5**
- If the integer is divisible by both 3 and 5, then it should print **divisible by 3 and 5**
- If none of the prior criteria are met, then the method should print the Integer element.

For example, if the collection contained the Integers 1, 2, 3, 4, 5, 10, 2, 4, 0, 15, 9, then the printout would look like:

1

2

divisible by 3

4

divisible by 5

divisible by 5

2

4

0

divisible by 3 and 5

divisible by 3

Once you have finished your project, open the terminal window in BlueJ and turn on record method calls. Create a new Numbers object, and then add at least eleven (11) Integers using the add method you wrote. Make sure you don't use the same collection as my example.

You must have an Integer that is divisible by 3, an Integer that is divisible by 5, and an Integer that is divisible by 3 and 5.

Demonstrate removing an element using the remove method you wrote, and then find the number of elements using your method. Finally, run your printNumbers method.

Copy all your calls into a text file and save it in your BlueJproject folder.

## Assignment Q5

### To be completed in week 5

#### **Part A:**

Imagine you need to write a program for a 24-hour clock with **hours, minutes** and **seconds**. Write a Java program in BlueJ with a method that prints all possible times the clock could display starting at 00:00:00 through to 23:59:59.

#### **Part B:**

Write a second method that takes three (3) parameters – **hours, minutes** and **seconds**. This method will print out all of the possible times the clock could display from one hour before the time passed to the method till one hour after the time passed to the method. For example:

If the method was passed the following values:

Hour = 11

Minutes = 23

Seconds = 44

The method would print all the times the clock could display from 10:23:44 until 12:23:44

## Submission

You should now have 4 BlueJ projects and 1 word document and a text file. You must zip the projects, the word document and the text file in into one zip file called usernameA1.zip. For example, mine would be ahendry10A1.zip

Submit this file via the Assignment 1 link on MySCU by the due date. Please leave enough time for it to upload, so do not submit at the last minute!