UNIVERSITY OF WESTMINSTER

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING
TIMED ASSESSMENT SEMESTER 1 2020/21

Module Code:	**5COSC001W**
Module Title:	**OBJECT ORIENTED PROGRAMMING – JAVA VERSION**
Module Leader:	**DR DIMITRIS DRACOPOULOS**
Release Time:	**12 January 2021, 10:00 (GMT)**
Submission Deadline:	**12 January 2021, 13:30 (GMT)**

Instructions to Candidates:

Please read the instructions below before starting the paper

- Module specific information is provided below by the Module Leader
- The Module Leader will be available during the exam release time to respond to any queries via the Discussion Board in the Assessment area of the module's Blackboard site
- As you will have access to resources to complete your assessment any content you use from external source materials will need to be referenced correctly. Whenever you directly quote, paraphrase, summarise, or utilise someone else's ideas or work, you have a responsibility to give due credit to that person. Support can be found at:

  https://www.westminster.ac.uk/current-students/studies/study-skills-and-training/research-skills/referencing-your-work
- This is an individual piece of work so do not collude with others on your answers as this is an academic offence
- Plagiarism detection software will be in use
- Where the University believes that academic misconduct has taken place the University will investigate the case and apply academic penalties as published in Section 10 Academic Misconduct regulations.
- ***Once completed please submit your paper via the Assignment content. In case of problems with submission, you will have TWO opportunities to upload your answers and the last uploaded attempt will be marked. Note that instructions on how to compile and submit your handwritten and/or typed solutions will have been sent to you separately.***
- ***Work submitted after the deadline will not be marked and will automatically be given a mark of zero***

Module Specific Information

IMPORTANT:
The exam paper is provided in two versions according to the Course you are enrolled on.

1) Java Version: BSc Computer Science, BSc Multimedia Computing, BSc Digital Media Development or BEng Software Engineering

2) C++ Version: BSc Computer Games Development

.

1. Implement the Java classes (with full details of fields and constructors) which are required to simulate the following problem:

A restaurant is located in a city, it has a specific address, a name, a menu (a number of daily dishes) and an unlimited number of customers visiting it every day. Each customer lives in a city, has a first name, a surname, a date of birth, an employer and exactly 3 favourite restaurants. Each city has a name and a number of restaurants.

[12 marks]

2. Briefly explain the meaning of the following Java access specifiers: *private*, *protected*, *public*. Which classes can access fields and methods declared with each of these?. If a class member omits the declaration of an access specifier which Java classes will be able to access it?

[8 marks]

3. Assume that class `Chair` is a parent class of class `ArmChair`. `Furniture` is a Java interface which is implemented by `Chair`. Justify which of the following statements are valid:

a) `Chair c1 = new ArmChair();`

b) `ArmChair c2 = new Chair();`

c) `Furniture c3 = new ArmChair();`

d) `ArmChair c4 = new Furniture();`

e) `Object c5 = new ArmChair();`

[5 marks]

4. **a.** Implement a class `Book` with just 3 private fields `title`, `author` and `num_of_pages`.

[2 marks]

b. Implement a single constructor of the class with the signature `Book(String, String, int)` initialising the title, author and num_of_pages fields with the values of the first, second and third argument respectively.

[2 marks]

c. Implement a getter method `getNumOfPages` which another class can call to access the private field num_of_pages.

[2 marks]

d. Implement a method `isLonger()` which accepts another `Book` object as an argument and returns true if the current object has more pages than the given book argument object.

[3 marks]

e. Override method `public String toString()` to return the title of the book and within parentheses the author, e.g. `Harry Potter(Rowling)`

[3 marks]

f. Implement class `TextBook` as a subclass of `Book`. The class should have a field `modules` which is an array of strings corresponding to the names of modules recommending this object as a textbook.

[2 marks]

g. Implement a single constructor with the signature:

```
TextBook(String title, String author, int num_of_pages,
         int num_of_modules)
```

The constructor should call the parent class constructor to initialise the inherited fields title, author, num_of_pages. The fourth argument num_of_modules should be used to create an array of num_of_modules elements and initialise with it the `modules` field.

The elements of the `modules` array should be initialised with objects containing the strings "Module1", "Module2", "Module3" etc. in total num_of_modules elements.

[5 marks]

h. Override method `toString` in `TextBook` so that it will return a string containing the title of the textbook, within parentheses the name of the author and following this the module names that recommend this textbook, e.g. `Big Java(Horstmann) Module1, Module2`.

You should NOT modify the access specifiers of the parent class fields title and author which should remain private.

[6 marks]

i. Implement a main method in which you create 3 objects of the `TextBook` class each one containing different data and display their contents on the screen by demonstrating how to utilise appropriately the `toString` method.

[2 marks]

5. Why Java provides two different ways of creating a thread? Justify your explanation with a very short example (you do not need to provide implementations of the `run()` methods).

[8 marks]

6. **a.** What is wrong with the following code which is supposed to display both "I am class T1" and an infinite number of "I am class C2" messages (not necessarily in this order)? Justify why this is happening:

```
class T1 extends Thread {
    public void run() {
        System.out.println("I am class T1");
    }
}


class C2 {
    public void run() {
        while (true)
            System.out.println("I am class C2");
    }
}

class T1Example {
    public static void main(String[] s) {
        T1 t = new T1();
        C2 c = new C2();

        c.run();
        t.start();
    }
}
```

[5 marks]

b. Without modifying class `T1` at all, describe (and provide the corresponding code) 2 different ways of fixing the problem. One of the two ways should be done without modifying at all class `C2`.

[16 marks]

7. Write a Java program to solve the following problem:

If the numbers 1 to 5 are written out in words: one, two, three, four, five, then there are $3 + 3 + 5 + 4 + 4 = 19$ letters used in total.

If all the numbers from 1 to 1000 (one thousand) inclusive were written out in words, how many letters would be used?

NOTE: Do not count spaces or hyphens. For example, 342 (three hundred and forty-two) contains 23 letters and 115 (one hundred and fifteen) contains 20 letters. The use of ”and” when writing out numbers is in compliance with British usage.

[19 marks]

END OF PAPER