Beijing-Dublin International College

AUTUMN TRIMESTER FINAL EXAMINATION - (2021/2022)

School of Computer Science

COMP2011J Object Oriented Programming

Dr. Rosemary Monahan
Assoc. Prof. Chris Bleakley
Dr. Seán Russell*

Time Allowed: 90 minutes

Instructions for Candidates:

Answer **All** questions.

BJUT Student ID:_____ **UCD Student ID:**_____

I have read and clearly understand the Examination Rules of both Beijing University of Technology and University College Dublin. I am aware of the Punishment for Violating the Rules of Beijing University of Technology and/or University College Dublin. I hereby promise to abide by the relevant rules and regulations by not giving or receiving any help during the exam. If caught violating the rules, I accept the punishment thereof.

Honesty Pledge:_____ **(Signature)**

Instructions for Invigilators

Non-programmable calculators are permitted.
No rough-work paper is to be provided for candidates.

Question 1: Basics of Java and OOP

a. Does Java use static or dynamic typing? Does Python use static or dynamic typing? What is the difference between the two? What are the benefits of static typing? **(10 points)**

b. Is Java a compiled or interpreted programming language? How does this process work? What are the benefits of this approach? **(10 points)**

c. Java has many rules about what names we can use for classes, methods and variables (known as identifiers). In addition to this programmers should also follow a number of conventions for identifiers. For each of the following, list the conventions (**not the rules**) that you should follow:

(i) All identifiers
(ii) Class identifiers
(iii) Variable identifiers
(iv) Constant identifiers **(10 points)**

d. What are the main differences between arrays in Java and Lists in Python? How can we find the size of an array in Java (Include an example of finding the size)? Give an example of declaring and constructing an array to remember 10 int values. **(10 points)**

e. Explain the difference between the two examples below. What is the output in each example?

```
int x = 120;
int y = x;
y++;
System.out.println(x + " " + y);
```

```
Date today = new Date(1,3,2019); // day, month, year
Date tomorrow = today;
tomorrow.incrementDay(); // increase day by 1
System.out.println(today.getDay() + " " + tomorrow.getDay());
```

(10 points)

(Question Total 50 points)

Question 2: Encapsulation, Interfaces and Inheritance

a. What is encapsulation? Why is it useful in Object-Oriented Programming? How do we implement encapsulation in Java? **(10 points)**

b. What is the function of the keyword static in Java? What effect does it have on a variable? What effect does it have on a method? Give an example of using the static method hello in the class Mess. The signature of the method is `hello(int n, String s)`
(10 points)

c. Describe how Interfaces can be used to enable code reuse in Java. Describe in detail how an interface would be defined and implemented to make reusable code. **(10 points)**

d. What is an immutable object? What changes must we make to a class to make its objects immutable? **(10 points)**

e. Explain the idea of **polymorphism** in your own words. You should illustrate using an example with some code showing its use. **(10 points)**

(Question Total 50 points)

Question 3: Nested Classes, Testing and Generics

a. What is a lambda function? When can we use a lambda function in Java? Define a lambda function to implement the `Calculator` interface defined below. The lambda function should calculate the product of these numbers multiplied by 2 (i.e $a * b * 2$). What parts of the syntax of a lambda function are optional?

```
interface Calculator {
    int performCalculation(int a, int b);
}
```

(10 points)

b. Given the following two examples of nested classes (a and b). Explain the difference between the two examples. Write the code required to create an object based on the nested class for each example.

```
public class C {
  public class D {

  }
}
```

(a)

```
public class E {
  public static class F {

  }
}
```

(b)

(10 points)

c. Describe the difference between traditional testing and unit testing. How do we know when we have enough code coverage in our testing?

(10 points)

d. Explain how generics improves type checking in the compiler. Give an example of a situation where an error would not be detected without generics. **(10 points)**

e. When does Java know what type a generic object will be using? Give an example of the code required to construct an object based on the generic class named GC. The object should use integer values for the parametrised type. The class has a constructor that takes no parameters.

(10 points)

(Question Total 50 points)

(Exam Total 150 points)