

NEWCASTLE UNIVERSITY

SEMESTER 2 2015/2016

ADVANCED PROGRAMMING

Time allowed - 3 Hours

Instructions to candidates:

Answer TWO questions from Section A,
and TWO questions from Section B.

The total marks available for this exam are 100.

Marks shown for sub-sections are indicative only.

[Turn Over]

SECTION A

Answer TWO questions in this Section.

Question A1

- a) Describe defensive copying in terms of a reference parameter to a constructor of a class. [2 marks]
- b) What is an immutable class? Give two advantages of immutable classes. [3 marks]
- c) Consider the following `Author` and `Book` classes:

```
public class Author {
    private String firstName;
    private String lastName;

    public Author(String firstName,
String lastName){
        this.firstName = firstName;
        this.lastName = lastName;
    }

    public String getFirstName(){return firstName;}

    public String getLastName(){return lastName;}

    public void setFirstName(String firstName) {
        this.firstName = firstName;
    }

    public void setLastName(String lastName){
        this.lastName = lastName;
    }
}
```

```
public final class Book {
    private final Author author;
    private final String title;
    private final Date datePublished;
                                   // java.util.Date

    public Book(Author author, String title,
                Date datePublished){
        this.author = author;
        this.title = title;
        this.datePublished = datePublished;
    }

    public Author getAuthor() { return author; }

    public String getTitle() { return title; }

    public Date getDatePublished(){
        return datePublished;
    }
}
```

- i) Explain the meaning of the `final` qualifier in the `Book` class declaration and in the declaration of its member fields. [3 marks]
- ii) Explain why instances of both `Author` and `Book` are mutable. [5 marks]
- iii) Re-write the `Book` class to make it immutable (without modifying the `Author` class). [8 marks]
- iv) Briefly describe the changes necessary to make `Author` immutable. If these changes were made, would it still be necessary to re-write the `Book` class? Briefly explain your answer. [4 marks]

Question A2

- a) Object factories are an alternative to constructors for the controlled instantiation of objects. Give an example of the application of the object factory design pattern in the standard Java libraries. [2 marks]
- b) List three advantages of an object factory compared with the use of constructors for object instantiation. [3 marks]
- c) Consider the following `ModuleCode` class.

```
public final class ModuleCode {  
    private final String prefix;  
    private final int number;  
  
    public ModuleCode(String prefix, int number){  
        this.prefix = prefix;  
        this.number = number;  
    }  
  
    public String getPrefix(){ return prefix; }  
    public int getNumber(){ return number; }  
}
```

- i) Explain why it is not necessary to define a separate interface for the `ModuleCode` class? [2 marks]
- ii) There are no constraints on the values of `prefix` and `number` passed to the constructor of the `ModuleCode` class. Suggest appropriate invariants with respect to the member fields and the exception(s) to throw if the parameters do not comply with the invariants. [3 marks]
- iii) Convert the `ModuleCode` class into a factory for `ModuleCode` objects. The factory must guarantee that each unique combination of `prefix` and `number` is represented by a single instance of the `ModuleCode` class (there can be no two `ModuleCode` instances with the same `prefix` and `number`). For example, the

factory will only ever provide a single `ModuleCode` object with prefix CSC and number 8002. Show your complete implementation of the `ModuleCode` class as a factory. Note: it is not necessary to define any new classes. [12 marks]

- iv) Do you need to override the `Object equals` method in your modified class? Briefly explain why. [3 marks]

Question A3

- a) Describe and illustrate with a diagram the structure of a doubly linked list. [5 marks]
- b) It is possible to use a resizable array or a doubly linked list to implement a list. Preferably using the Big 'O' notation, compare and contrast the performance of the two data structures for the following list operations:
- i) get or set an element at an arbitrary position in the list,
 - ii) add or remove an element at the head of the list, and
 - iii) insert or remove an element at an arbitrary position in the list (other than the head or tail).

In each case, explain any difference between worst-case performance for a resizable array and a linked list. [10 marks]

[CSC8002]

- c) Explain whether the worst-case performance of the following method is better for a collection implemented using a resizable array or a linked list.

```
void removeBelowLimit(Collection<Integer> vals,
int lim) {
    for (Iterator<Integer> i = vals.iterator();
        i.hasNext(); ) {
        if (i.next() < lim) {
            i.remove();
        }
    }
}
```

[5 marks]

- d) A sorted set uses a balanced binary tree to maintain its elements in their sorted order. The following comma-separated names are added to an empty sorted set of strings:

Mark, Elena, Stephen, Mark, Craig, Paul, Dave

Draw the state of the tree after the above sequence of additions. You may show intermediate states. [5 marks]

SECTION B

Answer TWO questions in this Section.

Question B1

- a) Describe and illustrate with a diagram the life cycle of a thread from being created to termination. List the ways by which a thread can be blocked and unblocked. [10 marks]
- b) Write a program that creates two threads, one to print out “Hip, hip!”, and the other to print out “Hooray!” (both in finite counting loops). After creating the two threads, the main program should terminate. [10 marks]
- c) Modify your program so that the thread printing out “Hip, hip!” would run first, and the thread printing out “Hooray!” be allowed to run only after the “Hip, hip!” thread has finished. [5 marks]

Question B2

- a) What are the three key features of a monitor and how are they used to support concurrent programming? [6 marks]
- b) A bridge over a river is only wide enough to permit a single lane of traffic. A collision can occur if two cars moving in different directions enter at the same time. We can call the cars moving in one direction “red cars” and the cars moving in the opposite direction “blue cars”. Write a monitor `SafeBridge` (with `redEnter`, `redExit`, `blueEnter`, `blueExit` operations) to control access to the bridge, so that the cars moving in different directions cannot concurrently use the bridge. [12 marks]
- c) Provide an implementation of `RedCar` class (or `BlueCar` class) describing the behaviour of red (or blue) car threads which access the safe bridge from part b). You may assume that a journey across the bridge takes 5 seconds. [7 marks]

Question B3

- a) What are the distinguishing characteristics of an event driven program as opposed to a conventional program? Provide an overview of the basic architecture of an event driven program, including a description of the event processing mechanism. What is the difference between an event handler and an event dispatcher? [8 marks]
- b) Describe briefly the basic steps that a Java programmer needs to take in order to set up event handlers and deal with events. [3 marks]
- c) You have been asked to implement a simple length conversion program as an event-driven program with a graphical user interface. The user is able to enter a length in miles into a text box. On pressing a button, the length entered is converted into kilometres and displayed in another text box (1 mile = 1.609 kilometre). Explain how you would construct such a program, by identifying the basic Java Swing components you would use, and specifying how events are handled. Provide outline code for the relevant parts of the application. [14 marks]

END