Two Hours

UNIVERSITY OF MANCHESTER
SCHOOL OF COMPUTER SCIENCE

M.Sc. in Informatics

High Level Programming (Java)

Date: Tuesday 19th January 2010

Time: 14.00 – 16.00

Answer ALL Questions in Sections A and B and one question from Section C

Electronic calculators may be used, provided that they cannot store text
SECTION A

Answer ALL questions in this section

Multiple Choice Section – not published.
SECTION B

Answer ALL questions in this section

B1. Write a class `LetterH` whose `toString` method produces a string depicting the letter “H” like this:

```
  *  *
  *  *
  *****
  *  *
  *  *
```

(Hint: remember to use escape sequences.)  (3 marks)

B2. Consider the following problem description: A user places coins in a vending machine and selects a product by pushing a button. If the inserted coins are sufficient to cover the purchase price of the product, the product is dispensed and change is given. Otherwise, the inserted coins are returned to the user. What classes should you use to implement it? Explain.  (3 marks)

B3. The following class `Lorry` is incomplete; a constructor and methods have yet to implement. Complete the class by implementing the constructor and four methods based on the specification described in comments.

```java
public class Lorry
{
    private int id;
    private double fuel;

    /**
     * Construct a specific lorry of some amount fuel.
     * @param aPlateNumber the plate number for this lorry
     * @param initialAmount the initial amount of fuel in the lorry
     */
    public Lorry(int aPlateNumber, double initialAmount)
    {
        fill in your code
    }

    /**
     * Get the plate number of this lorry.
     * @return the plate number
     */
    public int getPlateNumber()
    {
        fill in your code
    }

    /**
     * Fill fuel into the tank of this lorry.
     * @param amount the amount of fuel to add
     */
    public void addFuel(double amount)
    {
        fill in your code
    }
}
```

(B3 continues on the following page)
Consume fuel in the tank of this lorry.
@param amount the amount of fuel to use
*/
public void useFuel(double amount)
{
    fill in your code
}

/**
  * Get the state of fuel in the tank of this lorry.
  * @return the current amount of fuel
  */
public double readMeter()
{
    fill in your code
}

B4. A rectangle in the two-dimensional plane can be specified in various ways:
- by giving the top left corner (x,y) and its width and its height
- by giving the top left corner (x1,y1) and the bottom right corner (x2,y2)

Implement a class Rectangle of the field, left, top, width, height, with two
constructors, corresponding to the two cases above.
(Note: Your code needs to include two constructors only.)

B5. What is the output from the following program? List all printed results in order.

public class TestArray {
    public static void main(String[] args) {
        int[] ints = {22, 44, 66, 88};
        System.out.println("ints: " + ints);
        System.out.println("ints.length: " + ints.length);
        System.out.println("ints[2]: " + ints[2]);
        TestArray.print(ints);
        ints[2] = 99;
        System.out.println("ints[2]: " + ints[2]);
        TestArray.print(ints);
    }

    public static void print(int[] a) {
        for (int i=0; i<a.length; i++)
            System.out.print(a[i] + " ");
        System.out.println();
    }
}

(6 marks)
C1. A small hotel uses software written in JAVA to maintain a list of its current guests. This software includes three methods with the following signatures:

```java
static void enterGuest();
static void saveGuestList(),
static void readGuestList(),
```

These methods respectively allow the hotel to add a new guest to its list, to save the whole guest list to a hard disk, and read the existing list from the hard disk and put it in an array ready for updating.

These three operations are the only ones which are permitted to use or alter the guest list. They can be selected by typing a single character. The software uses a method with the signature:

```java
static char getCommand()
```

for prompting the user to type a character followed by RETURN, reading lines until a non-empty line is typed, and returning the first character on that line.

a) Write down a JAVA statement which obtains the next command, stores it in a variable `char command` and then executes an appropriate command, i.e.

- if the character is 'e' it will execute `enterGuest()``
- if the character is 's' it will execute `saveGuestList()```
- if the character is 'r' it will execute `readGuestList()```

(6 marks)

b) What alternative statement could you have used and why is your choice better? (2 marks)

c) What simple alteration could be made to your statement to allow a warning message if the user has typed an incorrect character? (2 marks)

d) What are the two simplest ways of allowing uppercase versions of the command, i.e., 'E', 'S' and 'R'? (2 marks)

e) What is an exception and what is the general form of the try statement which is used to handle exceptions? Illustrate your answer by showing how the method `getCommand()` might be written. (8 marks)
C2. a) What are the names of the three statements that JAVA uses for repetition? Write down the general form of each statement and explain what it does? (6 marks)

b) What criteria are used to choose between the above three statements? (3 marks)

c) Structured programming aims to reflect the structure of data in the statements which process it. Illustrate this principle by means of the following example.

The input to a program consists of a line of text which contains a possibly empty list of words which are separated by at least one space character. The words may be preceded and/or followed by spaces. The end of the line is marked by the asterisk character '★'. A word consists solely of letters and must contain at least one letter. No other characters besides spaces, letters and the asterisk may be present. You are given a method with the signature

\[
\text{char getChar()}
\]

which returns the next character from input.

Show how to design methods \text{skipSpaces}, to discard consecutive space characters, \text{readAndProcessWord}, to deal with words and a method \text{processLine}, to print the words in the line with one space following each word, and to count the number of words that the line contains. (7 marks)

d) Explain what differences one should make when other characters which must be ignored may appear in the line. In particular, explain any differences in the selection of a statement for repetition.

[Hint: You can make use of a method with the signature

\[
\text{public boolean isGarbage(char ch)}
\]

which returns true if \text{ch} is one of the characters to be ignored and false otherwise.] (4 marks)

END OF EXAMINATION