Autumn Examinations 2011/12

Module Title: Object-Oriented Programming 2

Module Code: **COMP7013**

School: Science & Informatics

Programme Title: BSc in Computing – Year 3
BSc (Hons) in Software Development – Year 2
BSc (Hons) in Software Development & Computer Networking – Year 2
BSc (Hons) in Web Development – Year 2
HDip in Science in Software Development

Programme Code: **KCOME_7_Y3**
KCOMP_7_Y3
KSDEV_8_Y2
KDNET_8_Y2
KWEBD_8_Y2
KSWDE_8_Y5

External Examiner(s):   Mr Peter Given
Internal Examiner(s):   Ms D. M. Dunlea
Mr Denis Long

Instructions: **Answer three Questions. Question 1 is mandatory.**

Duration: 2 Hours

Sitting: Autumn 2012

Requirements for this examination:

Section A – Question 1 (mandatory)

a)   Create a BankAccount class that holds an ID, name and balance for a user. An interest of 4.5% is associated with all bank account classes. The class allows the user to retrieve the name and balance in the account. The ID number can be returned. The user can deposit or withdraw money from the account. The account may be created with a 0 balance or a specified balance.   [10 marks]

b)   What modification would you make to the BankAccount class in order to implement the Measurable interface on it? The measure of a BankAccount can be considered to be its balance.   [6 marls]
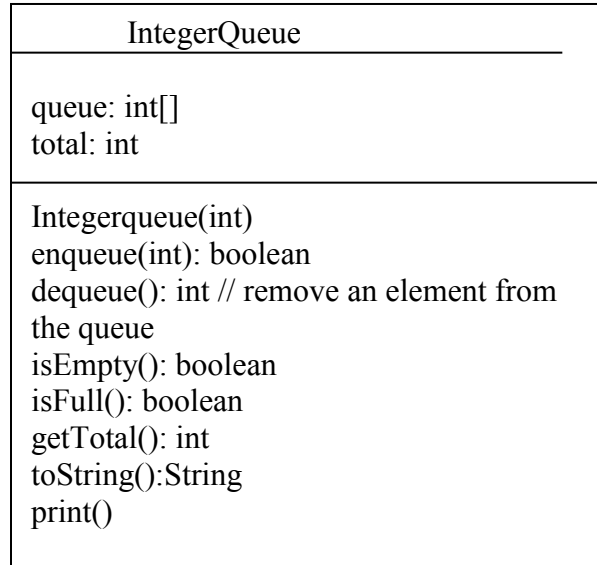
    public interface Measurable
    {
        double getMeasure();
    }

c)   Using the basic BankAccount class fulfil the following needs:
    A class called PersonalDetails gives the information about the owner of a BankAccount class.   These must include address and phone no and PRSI number. A BankAccount **cannot exist** unless these details are provided. The BankAccount also keeps track of the last transactions on that account. A Transaction class should be created to deal with this. A transacton may either be a **withdrawal**, a **deposit** or **interestAdded** at the end of the month. It should also contain the **amount** and the **balance remaining** in the account. Create these new classes and indicate any changes required to the original BankAccount class.   [24 marks]

Section B

Q.2

The following UML diagram represents an IntegerQueue class

IntegerQueue
queue: int[]   total: int

Integerqueue(int)
 enqueue(int): boolean
 dequeue(): int // remove an element from the queue
 isEmpty(): boolean
 isFull(): boolean
 getTotal(): int
 toString():String
 print()

Note: This queue is implemented using arrays. It stores positive integers. When removing an element from the queue you remove the first element placed in the queue. (FIFO)

a)   Write the complete code for this class.   [16 marks]

b)   Write a main program to create a queue to cater for 8 items. Using code enter 8 integers into the queue from 1 to 8. Try to enter a ninth item. Explain what happens in your code at this point?   [6 marks]

c)   Explain the following terms:
                    self-referential classes
                    dynamic memory allocation   [4 marks]

d)   What is a Layout manager? Describe the characteristics of the following layout managers:

        BorderLayout
        FlowLayout   [4 marks]

Q3

a) Explain what is meant by the term object serialization with emphasis on what needs to be saved and what doesn't? What word prevents saving certain information? [8 marks]

b) What happens during the process of deserialization? [8 marks]

c) Write the code for reading a Text File. [10 marks]

d) Explain giving an example why we would use interfaces. [4 marks]

Q4

a) What do you understand by the term event-driven? Explain what an inner class is. Why are they useful when writing event-handlers? [6 marks]

b) You are required to develop the class IntegerSet. Each IntegerSet object can store integers in the range 0-10.The IntegerSet class uses an array of Boolean values to record what integers are in the set. For example, array element a[i] is true if integer i is in the set, array element a[j] is false if integer j is not in the set. The no-argument constructor initializes the Java array to the "empty set" (i.e. a set whose array representation contains all false values).

Provide the following methods:

- Method union creates a third set that is the union of two existing sets (i.e. an element of the third set's array is set to true if that element is true in either or both of the existing sets).
- Method intersection creates a third set that is the intersection of two existing sets (i.e. an element of the third set's array is set to false if that element is false in either or both of the existing sets).
- Method insertElement inserts a new integer k into a set (by setting a[k] to true).
- Method deletElement deletes integer m (by setting a[m] to false)
- Method toString returns a string containing a set as a list of numbers separated by spaces. Include only those elements that are present in the set. Use '---' to represent an empty set.
- Method isEqualTo determines whether two sets are equal.

- Method isInSet determines whether or not a specified integer is in the set. Returns a Boolean.

[18 marks]

c)   Write a program to test class IntegerSet. Test that all your methods work correctly.

[6 mark]