Athlone Institute of Technology

School of Engineering

Semester 1 Examinations 2016

December Session

Bachelor of Science in Software Design (Game/Cloud Development)

Year 3

Software Development

External Examiner(s): Dr. Stephen Davy
Mr. Jerh O Connor

Internal Examiner(s): Dr. Enda Fallon

Instructions to candidates:
Read all questions carefully.
All questions carry equal marks.
Answer **Three** out of **Four** questions.

Time Allowed: 2 Hours

No. of pages including cover sheet: 5

Q.1. (a) The following table stores details of members of a gym including membership number, membership type, membership cost, when the member first registered and account balance.
Assume that a class called Member **already exists** which encapsulates the data for member records including all necessary set and get methods.

Write a Main class which uses a collection to create and store the objects below.

(6 Marks)

Once the necessary objects have been created:

- Calculate the total number of members who use the gym
- Calculate the average membership cost for all members
- Print the names of any members with overdue balances.
- Gym management wish to introduce a 20% loyalty discount for members who have been with the gym for more than 5 years. Based on the members below, calculate how much the loyalty scheme would cost the gym?

(6 Marks)

Membership Number	Membership Type	Membership Cost €	Member Since	Account Balance
4243	Gym and Swim	730	2012	0
5626	Gym and Swim	730	2011	-300
6262	Gym Only	560	2009	0
7373	Gym Only	560	2010	-560
7288	Fitness Class	280	2008	280

(b) Outline the updates required to the Member and Main classes in part (a) in order to

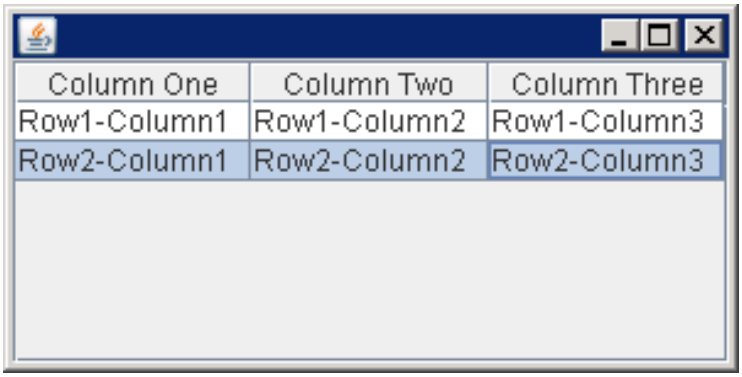
1. Serialize the objects to a file called “member.ser”
2. Deserialize the contents of “member.ser”

(8 Marks)
[20 Marks]

Q.2. (a) With regard to Javas platform independence, discuss using a diagram the performance implications of choosing Java as a programming language. In your discussion illustrate the implications of Javas compilation process on performance.

(8 Marks)

(b) Consider the following Graphical User Interface (GUI) which displays a table using a Java JTable component with a TableModel object.



Using sections of skeleton code outline how to implement the functionality of this GUI.

(12 Marks)
[20 Marks]

Q.3. (a) "A collections framework is a unified architecture for representing and manipulating collections." With reference to the Java Collections Framework (JCF) explain the need for a "unified architecture". Your explanation should outline why the JCF was introduced. What are the major elements of the JCF?

(7 Marks)

(b) With reference to the JCF, explain using relevant sections of code the term iterator. In your explanation include (a) the purpose of an iterator (b) how to access an iterator (c) types of iterator.

(7 Marks)

(c) Outline using short sections of relevant code the operation of the list interface in the JCF. What is the purpose of the interface? What is the package hierarchy of the interface? What differentiates the List interface from other JCF features such as maps?

(6 Marks)
[20 Marks]

Q.4. (a) Briefly explain the term “Design pattern”. In your explanation include a description of (a) what a design pattern is (b) how design patterns evolved (c) name the elements of a design pattern.

(5 Marks)

(b) “An interface declares methods but does not implement them. All the methods are implicitly public and abstract”

Explain this statement using a diagram and code examples. In your explanation describe (a) the purpose of interfaces (b) why an interface is a very abstract class (c) why an interface would be used in preference to an abstract class.

(6 Marks)

(c) Using Java RMI, re-factor the following application so that we can access the Customer objects remotely.

```
Class Customer{
        private String theName;
        private String theAddress;

        public Customer (String aName, String aAddress)
        {
           theName = aName;
           theAddress = aAddress
        }

        public String getName()
        {
           return theName;
        }
        public int getAddress()
        {
           return theAddress;
        }
    }
```

```
public class CustomerServer
{
    public static void main(String [] args)
    {
       Customer Tom = new Customer("Tom Hynes", “11 High Road”);
       Customer Ann = new Customer("Ann Jones", “10 Low Road”);
       System.out.println( "Customer Name: " + Tom.getName() +  “ lives at “ +
                                                          Tom.getAddress());
    }
}
```

(9 Marks)
[20 Marks]