

0115

00053

**THE OPEN UNIVERSITY OF SRI LANKA
DEPARTMENT OF ELECTRICAL & COMPUTER ENGINEERING
BACHELOR OF SOFTWARE ENGINEERING
ECI4262 - OBJECT ORIENTED DESIGN AND PROGRAMMING
FINAL EXAMINATION – 2014/2015**



CLOSE BOOK TEST

Date: 12th of September 2015

Time: 09:30 – 12:30 hrs

INSTRUCTIONS TO CANDIDATE:

- This paper consists of two (2) sections SECTION A and SECTION B in four (4) pages.
- SECTION A has one (1) compulsory question.
- SECTION B has five (5) questions, the total marks allocated to each question is equal in this section. Answer any three (3) questions.
- Assume reasonable values or any suitable assumptions for any data not given in or if any doubt as to the interpretation of the wording of a question. Clearly state such assumptions made on the script.
- You are NOT allowed to use any study material or any other electronic resource during the examination.

SECTION A

Answer all parts given in **Q1** in this section.

[Q1] (Compulsory Question) [40 Marks]**Credit Card Processing - Functional Requirements:**

Background: The credit card payment service is one of the core businesses in banking industry. The usage of credit cards is very popular all over the world, even on the virtual community - Internet. YBK Bank is a global commercial bank based in Colombo. It plans to build an online credit card management system in order to handle such a large amount of information in an efficient way and provide better service to their customers. Most commercial banks provide the following major online services to facilitate their credit card business: credit card application; online credit card payment and special offers to customer e.g. Cash Dollars; credit card transaction checking etc. With the advance of information technology, YBK bank is planning to extend the above services through the Internet. The information systems development team of YBK bank is going to conduct the feasibility analysis, system analysis and design of the online Credit Card Management System (CCMS). For the simplification of this case study, let's have a look at only two of the major functional requirements.

Application for Credit Card: In order to obtain a YBK bank credit card, customers may make the application by filling up an online application form through the CCMS. In the application, they need to specify the type of credit card that they want to apply, e.g. Platinum, Gold, or Standard. The differences among them are the requirement on the minimum personal annual income, credit limit and annual fee. Information, such as personal contact details, current employment, and financial status, is also required in the application. Once YBK bank accepts the application, the bank will send a confirmation mail to the applicant and state the available date, expiration date and credit limit of the credit card. When the applicants receive the mail with the credit card, they need to call the CCMS to activate the card with their NIC and credit card numbers.

Produce Monthly Statement: At the midnight of statement date in each month, CCMS generates monthly electronic statements for each customer. Paper-based statements are also produced for the customers who prefer to receive them. For each credit card CCMS computes the amount of debts and minimum return by considering the total transactions made within the month. The minimum amount of return should be 4% of the total debts. Late payment fee will be charged by

0.05% of the total debts plus extra Rs.1000. If the payment is received by auto-pay, the minimum amount of payment will be cleared automatically. If paper-based statement is used, an extra Rs.100 service fee will be charged. Finally, monthly statement is recorded and mailed to the cardholders who selected paper-based monthly statement.

Examine the above case study and answer the following questions.

- a) Identify four use cases and give a high level description of each [16 marks]
- b) Draw a use case diagram for the use cases identified in part(a) above. [18 Marks]
- c) Explain the use of <<extends>> and <<includes>> with respect to use cases. [06 marks]

SECTION B

Answer **any three (3)** questions in this section.

[Q2] [20 Marks]

- a) Explain the concept of garbage collection in Java. [04 marks]
- b) When is the garbage collector called? [02 marks]
- c) Explain the concept of packages in Java. How are they used? [04 marks]
- d) Explain the terms over-loading and over-riding using examples. [10 Marks]

[Q3] [20 Marks]

- a) State **three** reasons for why the design patterns are important in object-oriented software development. [06 marks]
- b) What are the **three** main categories of design patterns [06 marks]
- c) Explain the use of following design patterns [08 marks]
- Singleton Pattern
 - MVC Pattern

[Q4] [20 Marks]

- a) Explain *pass-by-value* and *pass-by-reference* in java. Write a short segment of code to demonstrate the difference. [10 Marks]
- b) What is a Java Interface? Give an application example of using an Interface. [06 marks]
- c) What is the difference between the “==” operator and an *equals* method in java? [04 marks]

[Q5] [20 Marks]

- a) What does the phrase “Write Once, Run Anywhere” mean, in reference to Java? How is this possible? [04 marks]
- b) Explain the use of *static methods* and *static variables* in classes in Java programs. Give an example containing code for each. [08 marks]
- c) Write a java class named **Car** and add the following. [08 marks]
- An instance variable called **name** of type **String**.
 - A constructor that accepts a string parameter and sets the car's name using this parameter.
 - A *getName()* method that returns the car's name.

[Q6] [20 Marks]

- a) A use case for a university registration system is briefly described as follows.
The name of the use case is “Register a course”. In this use case, a student requests to register a course. The system then checks whether the student’s prerequisites are satisfied. If so, the system will let the student register and record the registration. Otherwise, the system notifies the student that he/she cannot register for the particular course.
Identify the *classes* used in this use case. Draw a class diagram to give the names of the classes, the attributes and operations for each class. You just need to give the attributes and operations related to this use case. Then give the associations and multiplicities of the associations. [12 marks]
- b) Describe the following relationships in a class diagram. [8 marks]
- Association
 - Aggregation
 - Composition
 - Generalization