UNIVERSITY *of* LIMERICK

OLLSCOIL LUIMNIGH

COLLEGE *of* INFORMATICS *and* ELECTRONICS
Department of Computer Science and Information Systems

End of Semester Exam Paper

Academic Year:	2005/2006	**Semester:**	Spring
Module Title:	Software Testing & Inspection	**Module Code:**	CS4004
Duration of Exam:	2.5 Hours	**% of Total Marks:**	60
Lecturer:	N. Power and A. McElligott	**Marks out of :**	100

Instructions to Candidates:

Section A: **ALL** questions should be attempted in this section.

Section B: You are expected to attempt **2** questions from this section.

State clearly any assumptions you make.

Q1 Black-box Testing **(20 marks)**

To enter a radio competition an entrant must text a message of the format

Answer Forename Surname

e.g., B Joe Black

A Java program processes a stream of these text messages. An administrator will run this program entering the correct answer to the competition, i.e., either A, B, or C. For example,

Java TextStreamer B

This program only stores messages that are of the correct format and contain the correct answer. This program expects that on any occasion there will be at least 50 such messages. From these messages this program randomly selects a competition winner.

You are required to design test cases using equivalence classes and boundary value analysis. The test cases should be documented as follows:

(i) for each equivalence class you identify you should specify its number, its description, whether its is valid/invalid and provide an example.

(ii) a table specifying for each test case its number, the test case (i.e., the input value), whether it is valid or invalid, classes covered (including boundaries if any), and expected outcome.

Q2 White-box Testing **(20 marks)**

The program in Figure A2.1 begins by creating a FileReader object and a BufferedReader object to read the file terms.txt. This comma-separated data file contains technical terms in five languages namely, English, Irish, Italian, Spanish and French and is in alphabetical ascending order on the English language entries. The following are two sample entries from this file.

OK, Ceart go leor, Attendere, Aceptar, OK

Welcome, Fáilte, Benvenuti, Bienvenido, Bienvenue

This program allows the user to enter a word or phrase (it is assumed that this entry is a word or phrase of the English language). The program returns all equivalent translations of this word or phrase if it exists in the data file. For example, if the user enters the word *welcome* the program will display output similar to that shown in Figure A2.2. An appropriate message is displayed if the word or phrase is not found in this data file.

(a) **[16 marks]**

For the program in Figure A2.1 you are required to complete the following:

(a) Write test cases to achieve 100% statement coverage of this program. For each test case you should write its test case number, its description, expected outcome and actual outcome.

(b) Draw a Control Flow Graph (CFG) for this program.

(c) Using your CFG write test cases to achieve

(i) 100% decision/branch coverage and

(ii) 100% condition coverage.

For each test case you should write its test case number, its description, expected outcome and actual outcome. In your answer you should identify whether a particular test case concerns decision/branch testing or condition testing.

```
1  import javax.swing.JOptionPane;
2  import java.io.*;
3  public class TranslationOfWordOrPhrase
4  {
5    public static void main(String []  args) throws IOException
6    {
7      FileReader aFileReader = new FileReader("terms.txt");
8      BufferedReader aBufferReader = new BufferedReader(aFileReader);
9      String searchText, lineFromFile, output = "";
10     String [] termsArray;
11     boolean found = false;
12     searchText = JOptionPane.showInputDialog(null,"Enter a word/phrase");
13     while (((lineFromFile = aBufferReader.readLine()) != null) && !found)
14     {
15       if (lineFromFile.toLowerCase().startsWith((searchText + ",").toLowerCase()))
16       {
17         termsArray = lineFromFile.split(",");
18         output = termsArray[0].trim() + " in English translates to:\n";
19         output += termsArray[1].trim() + " in Irish\n";
20         output += termsArray[2].trim() + " in Italian\n";
21         output += termsArray[3].trim() + " in Spanish\nand ";
22         output += termsArray[4].trim() + " in French";
23         found = true;
24       }
25     }
26     aBufferReader.close();
27     aFileReader.close();
28     if (found)
29       JOptionPane.showMessageDialog(null, output, "Translations",
30         JOptionPane.INFORMATION_MESSAGE);
31     else
32       JOptionPane.showMessageDialog(null, "Word/phrase does not exist on file");
33     System.exit(0);
34   }
35 }
```

Figure A2.1: Program code for question on white-box testing

Figure A2.2: Sample output shown if word or phrase is found in the data file

(b) **[4 marks]**

Explain decision/condition testing paying particular reference to the program in Figure A2.3.

```
1  import javax.swing.JOptionPane;
2  public class DecisionConditionTesting
3  {
4    public static void main(String[] args)
5    {
6      String vowels = "aeiou", response;
7      do
8      {
9        String c = JOptionPane.showInputDialog(null, "Enter a character");
10       if (vowels.indexOf(c.toLowerCase()) != -1)
11         JOptionPane.showMessageDialog(null,
12                           "Character entered is a vowel");
13       else
14         JOptionPane.showMessageDialog(null,
15                           "Character entered is not a vowel");
16       response = JOptionPane.showInputDialog(null,
17                           " Check another character (y/n)");
18     } while (!(response.equals("n") && !(response.equals("N"))));
19     System.exit(0);
20   }
21 }
```

Figure A2.3: Program for part **(b)** of Q2

Q3 Bug Report **(10 marks)**

A system tester has filed the following bug report.

> "Another whopper today, is this Friday the 13th? I tried to enter a time in the time field in the format hh:mm. While this program should accept a time in 24-clock format IT DOESN'T."

Comment on this bug report paying particular attention to the elements of a bug report you would expect a professional system tester to produce.

End of Section A

Section B: Attempt 2 Questions

Q4 Bugs, Bug Reporting and Bug Tracking (25 marks)

(a) Identify the users of a professional bug report. For what purposes would each user you have identified use such a report. **[9 marks]**

(b) Distinguish between the priority and the severity of a bug. **[7 marks]**

(c) Briefly describe the life cycle of a bug in terms of its possible states. **[9 marks]**

Q5 Software Inspections (25 marks)

(a) Distinguish between a walkthrough and an inspection paying particular attention to
- (i) personnel involved and their roles and
- (ii) the purpose of each. **[10 marks]**

(b) What elements would you expect to find in the checklist for
- (i) a requirements inspection and
- (ii) a program inspection? **[9 marks]**

(c) List 4 differences between inspection and testing. **[6 marks]**

Q6 Types and Levels of Testing (25 marks)

(a) Distinguish between black-box testing and white-box testing. **[5 marks]**

(b) Why is it necessary to do both types of testing? Outline your reasons and support your answer with an example. **[10 marks]**

(c) Distinguish between unit testing, integration testing and system testing discussing the need for different levels of testing. **[10 marks]**

Q7 Mix of Topics (25 marks)

(a) How can the following specification be improved upon? **[10 marks]**
A company transmits all their data as four-digit integers. A program is required that requests a series of four-digit integers and encrypts this sequence as follows: replace each digit by (the sum of that digit plus 7) modulus 10. Then swap the first digit with the third, and swap the second digit with the fourth. Appropriate messages should be displayed.

(a) Define risk. What is the relationship between risk and the need for testing? **[10 marks]**

(b) What is the purpose of a CRUD matrix in testing? **[5 marks]**

End of Section B