UNIVERSITY *of* LIMERICK

OLLSCOIL LUIMNIGH

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

End of Semester Exam Paper

Academic Year:	2006/2007	**Semester:**	Spring
Module Title:	Software Testing & Inspection	**Module Code:**	CS4004
Duration of Exam:	2.5 Hours	**% of Total Marks:**	70
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 Functional Testing **(20 marks)**

To receive assistance when attempting to complete a crossword a person can send a text message containing the crossword type followed by the clue identifier e.g., C 7d (which denotes Crossaire 7 down). There are two crossword types, Crossaire and Simplex which are abbreviated to C and S, respectively. A Java program exists that processes these text messages, one at a time.

In response, the texter receives a message produced by this program containing the answer to this clue for today's crossword. You may assume that this program accesses the correct answer file for today. If the crossword type and/or the clue identifier are invalid, appropriate error messages are displayed. The valid numeric aspect of the clue identifier will be in the range 1 to 32.

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 create 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 Structural Testing **(20 marks)**

A pangram is a sentence that contains each character of an alphabet in either uppercase or lowercase form. The program in Figure A2.1 reads each line of a file named story.txt. If any line on this file is a pangram of the English alphabet a counter is incremented. A count is also kept of entries that are not pangrams (cf. Figure A2.2).

(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 state whether a particular test case concerns decision/branch testing or condition testing.

```
1  import javax.swing.JOptionPane;
2  import java.io.*;
3  public class PangramsOnFile
4  {
5    public static void main(String[] args) throws IOException
6    {
7      FileReader aFileReader = new FileReader("story.txt");
8      BufferedReader aBufferReader = new BufferedReader(aFileReader);
9      String lineFromFile = "", outputPrompt = "";
10     int pangramCount = 0, notPangramCount = 0;
11     boolean pangram;
12     while ((lineFromFile = aBufferReader.readLine()) != null)
13     {
14       lineFromFile = lineFromFile.toUpperCase();
15       pangram = true;
16       for (char i = 65; i <= 90 && pangram; i++)
17         if (lineFromFile.indexOf(i) == -1)
18           pangram = false;
19       if (!pangram)
20         notPangramCount++;
21       else
22         pangramCount++;
23     }
24     aBufferReader.close();
25     aFileReader.close();
26     outputPrompt = "Of the " + (notPangramCount + pangramCount) + " entries ";
27     outputPrompt += "read from file " + pangramCount + " are pangrams.";
28     JOptionPane.showMessageDialog(null, outputPrompt, "Output",
29        JOptionPane.INFORMATION_MESSAGE);
30   }
31 }
```

Figure A2.1: Program code for question on Structural Testing

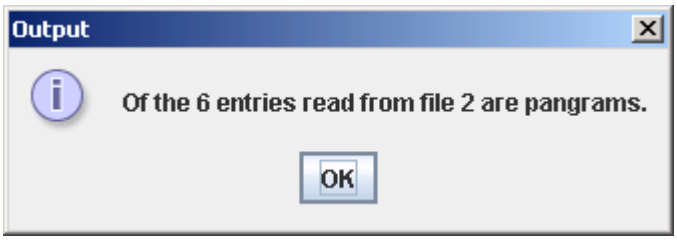

Figure A2.2: Sample output

(b) **[4 marks]**

Explain decision/condition testing. Support your answer with an example.

Q3 Software Testing Body of Knowledge **(10 marks)**

Compare and contrast the scope of CS4004 with the scope of the SWEBOK KA Software Testing.

End of Section A

Section B: Attempt 2 Questions

Q4 Software Inspections (25 marks)

(a) Distinguish between static and dynamic testing. **[9 marks]**

(b) What are the advantages of static testing over dynamic testing? **[7 marks]**

(c) "An inspection meeting is really a collection meeting." Explain this statement, referring to the procedure that is used for an inspection, paying particular attention to the preparation. **[9 marks]**

Q5 Testing Tools and Automation (25 marks)

(a) "Tools can assist the tester to do manual testing more efficiently or can be used to carry out automated testing." Briefly describe two special-purpose tools than can assist with manual testing. **[7 marks]**

(b) Discuss the advantages of automation for system-level testing? **[8 marks]**

(c) Under which circumstances should the testing of modern software product **not** be automated? Give reasons for your answers. **[10 marks]**

Q6 Testing Paradigms (25 marks)

(a) Explain the rationale of Context-Driven Testing, distinguishing it from other paradigms of testing. **[9 marks]**

(b) What are the key features of exploratory testing? How does it differ from random testing? **[7 marks]**

(c) Comment on the relationship between exploratory testing and bug reporting. **[9 marks]**

Q7 Reporting and Categorising Errors (25 marks)

(a) Bug Type is one essential element of a professional Bug Report? What are the others? **[9 marks]**

(b) Outline one particular scheme for categorising errors or bugs and comment on its usefulness. **[9 marks]**

(c) Discuss the different users of Bug Reports? **[7 marks]**

Q8 Paper titled "What Do We Know About Defect Detection Methods" (25 marks)

(a) Define the terms 'Defect', 'Failure' and 'Error', outlining the relationships between them. **[7 marks]**

(b) Distinguish between the efficiency and the effectiveness of a 'defect detection method'. Your answer should refer to how each of these is measured? **[9 marks]**

(c) Summarise the advantages and disadvantages for different types of software artefacts of the two defect detection methods. **[9 marks]**

End of Section B