



**OXFORD BROOKES UNIVERSITY**

**BACHELOR OF SCIENCE (HONOURS)**

**APRIL 2009 EXAMINATION**

**15<sup>th</sup> APRIL 2009**

**U08182 INFORMATION SYSTEMS DESIGN**

**TIME : 2 Hours + 10 Minutes Reading**

**NUMBER OF PAGES : 1 Cover Sheet and 7 Pages of Questions**

**INSTRUCTIONS:**

- ☐ **SECTION A QUESTION IS COMPULSORY.**
- ☐ **Answer any TWO questions from SECTION B.**
- ☐ **Please start every question on a new page.**
- ☐ **Answers will not be marked if they are illegible.**
- ☐ **Enter the question numbers (in the order you have attempted) in the boxes provided in the answer script.**
- ☐ **Write your INDEX NUMBER and MODULE NUMBER on the cover page of the answer script.**



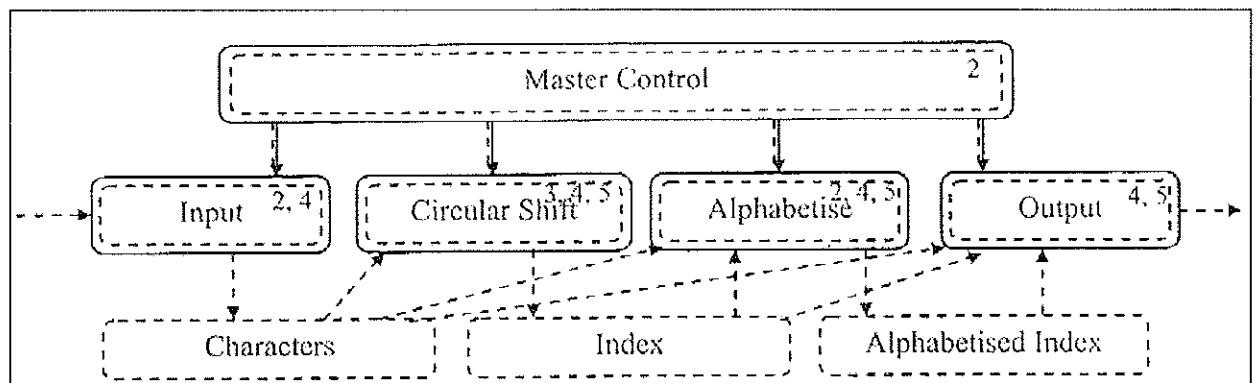
## SECTION A

(THIS QUESTION IS COMPULSORY)

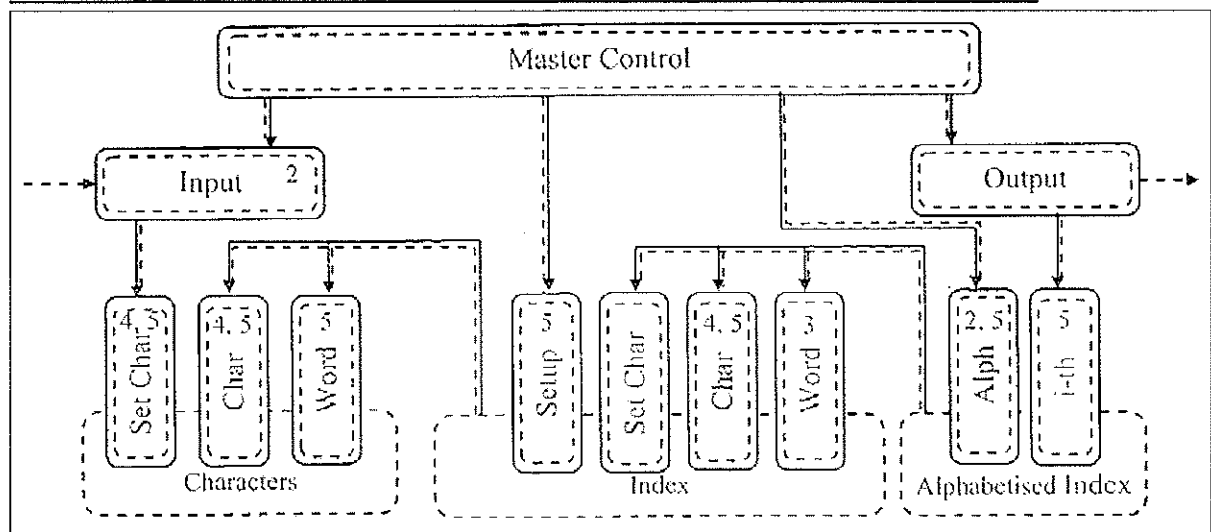
### QUESTION 1

- a) The Keyword in Context (KWIC) index system accepts an ordered sequence of lines of text. Each line is an ordered sequence of words, and each word is an ordered sequence of characters. A line might be circularly shifted by repeatedly removing the first word and appending it at the end of the line. The KWIC index system outputs a listing of all circular shifts of all lines in alphabetical order. Below is the revealing scenario interaction based on scenario evaluation.

#### Revealing Scenario Interaction for Design 1 based on quality requirement



#### Revealing Scenario Interaction for Design 2 based on quality requirement



- i) By using the Software Architecture Analysis Method (SAAM), evaluate both designs by completing the following table:

[8 marks]

Scenario			Modification Cost	
No.	Description	Weight	Design 1	Design 2
1	To operate in the batch fashion	10%		
2	To operate in an incremental fashion	20%		
3	To eliminate noise words in shifted lines	15%		
4	To change the internal representation of lines	25%		
5	To change the internal representation of intermediate data structures.	30%		
Overall				

- ii) Compare the two candidate designs to see which one satisfies its quality requirement on modification better.

[4 marks]

- b) Software Testing is an empirical investigation conducted to provide stakeholders with information about the quality of the product or service under test, with respect to the context in which it is intended to operate. This includes, but is not limited to, the process of executing a program or application with the intent of finding software bugs.

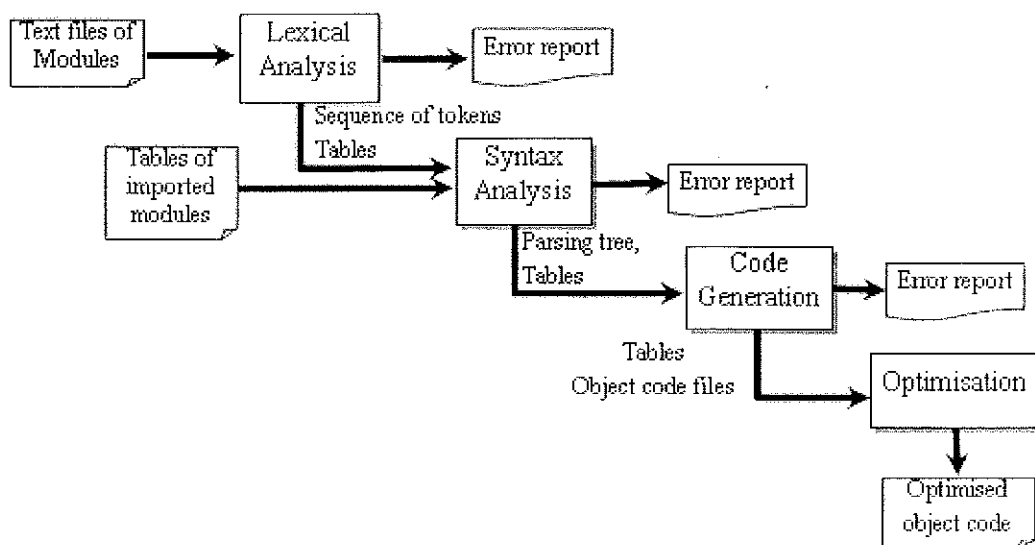
- i) Testing a system reveals no errors. What conclusion can be drawn from this?

[3 marks]

- ii) Testing a system reveals a few errors. What conclusion can be drawn from this?

[3 marks]

- c) In a compiler, the lexical analysis phase reads in the source code of the modules to be compiled, which is in the form of a sequence of characters stored in a text file. It generates a table of identifiers and a sequence of tokens, where each token represents a lexical element in the source code. The outputs are stored in a number of files. These files together with files previously generated in the compilation of the imported modules are passed to the syntax analysis phase. The syntax analysis phase parses the sequence of tokens into a parsing tree and generates some more tables. For example, there could be a table to associate variables with their types. These outputs are also stored in files for consumption by the code generation phase. The code generation phase reads in these files and produces object code and more tables, such as tables providing information about the exported procedures, variables and types for the linker program. These outputs are also stored in files. The optimization phase is optional, which improves the efficiency of the object code. Of course, in each phase, errors in the source code of the program may be discovered. In such cases, an error report may be generated and either displayed on the screen or written into an error report file. The compilation will terminate after an error is discovered.



- i) List and explain the software architecture notation used in the architecture style. [6 marks]

- ii) Use software architectural visual notation to describe the architecture of compilers.

[10 marks]

**[TOTAL MARKS FOR QUESTION 1: 34 MARKS]**

## SECTION B

(Answer any TWO Questions)

### QUESTION 2

- a) Consider the following program segment:

```
counter, result: integer
1. counter = 0;
2. result = 0;
3. WHILE counter < 5
4.     CASE counter IS
5.         2: result = counter;
6.           counter = 4;
7.         3: result = counter;
8.           counter = 4;
9.     END CASE
10.    counter = counter + 1;
11. END WHILE
```

- i) Draw the Flow Graph that models the code segment.

[8 marks]

- ii) McCabe Cyclomatic complexity is software metric that provides a quantitative measure of the logical complexity of a program. Calculate the McCabe's Cyclomatic Number to find out the total number of independent paths.

[2 marks]

- iii) Identify each independent path.

[4 marks]

- iv) Find a set of test data that tests each path? Explain your answer.

[6 marks]

- b) The following is a description of a use case for lecturers to select modules to teach at a university department.

<b>Actor: Lecturer</b>	<b>The system</b>
The lecturer enters his/her username and password.	The system verifies that the current password is valid, and prompts to select a semester.
Lecturer enters a semester number.	The system prompts the lecturer to select from a list of modules to be delivered in the semester - that have not been assigned to another lecturer.
The lecturer enters his/her module selection.	The system notifies the lecturer that the module selection is recorded.

- i) Identify the input and output variables of the system, and the data to be stored in the system using the first column in the table below:

**[7 marks]**

Test data for the Select Module use case

<b>Variables</b>		<b>Test data</b>
Input		
Output		
Stored data		

- ii) Make a concrete scenario from the above use case description, and then derive test data from the scenario and fill the second column of the table above.

**[6 marks]**

**[TOTAL MARKS FOR QUESTION 2: 33 MARKS]**

### QUESTION 3

- a) Imagine your company has decided to introduce a software product that will track the inventory, handle the sales (e.g., authorize credit/debit card payments), and generate reports about the inventory and sales for a collection of small independent toy stores. Your company expects there to be many updates to the system's functionality. The targeted end-users (toy stores and their employees) are not knowledgeable about computers, are spread across Canada, and would like to have a fast turn-around on updates to the functionality of the system.
- i) Explain why client-server architecture is suitable for this design. **[6 marks]**
- ii) Sketch a client-server architecture for the new software product described above. What will be the major components? How will the components be connected? What functionality will be provided by each component? **[8 marks]**
- b) Discuss how software architecture affects software performance. **[8 marks]**
- c) The United Services Athletic Club has asked you to develop a computer based information system for their records. The system has to register new members and allocate each member to a particular category: Competitive Sports (such as swimming or running), Court Sports (such as tennis or badminton) and Team Sports (such as cricket, soccer, rugby, and hockey). Every two months, for each of the team sports, the system schedules fixtures (games) against the other seven Athletic Clubs in the surrounding area. For the Court Sports, the system enables members to book a court, or to cancel an existing booking. For the competitive sports, the system arranges events every month where club members can compete against each other. In addition to all of these activities the system must also keep a record of individual members' achievements, such as speed over 100 meters, or scores from court based matches or team matches.
- i) Identify at least five use cases for this system and draw one use case diagram showing all of your use cases together with the actors involved with those use cases. **[11 marks]**

**[TOTAL MARKS FOR QUESTION 3: 33 MARKS]**



## QUESTION 4

A text consists of a sequence of paragraphs separated by the 'new line' symbol; each paragraph consists of a sequence of words that are separated by spaces and punctuation symbols such as ',' and '.'. Each word is a sequence of letters. A paragraph can contain an arbitrary number of characters and letters. To display the text on screen or to print the text on pages, the text must be wrapped. That is, the paragraphs must be decomposed into a number of lines to fit into the width of the screen or page. Such decompositions should not break in the middle of a word. Each line should contain as many words as possible.

Assume that all letters and characters take the same size of space when displayed or printed; design the architecture of a program that performs the text wrapping, by addressing the following issues.

- i) Present your design of the architectural structure using Software Architectural Visual Notation.

[8 marks]

- ii) Describe the functions and properties of each component and connector in your design.

[10 marks]

- iii) Discuss the necessary modifications to the architecture and components of your design if the letters and characters in a text may take variable sizes of space when displayed on screen or printed on paper. (If new component(s) and/or connector(s) are introduced, also describe their functions and properties.)

[7 marks]

- iv) Give the reasons why the architectural style that you selected is appropriate.

[4 marks]

- v) On what basis does a software developer select the appropriate architectural styles for a software system under construction?

[4 marks]

**[TOTAL MARKS FOR QUESTION 4: 33 MARKS]**

**- END OF PAPER -**