UNIVERSITY *of* LIMERICK

OLLSCOIL LUIMNIGH

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

Assessment Paper

Academic Year: 2005/2006 **Semester:** 2
Module Title: Business Oriented Programming Languages **Module Code**: CS4556
Duration of Exam: 2.5 hours **% of Total Marks:** 60%
Lecturer : Michael Coughlan **Paper marked out of :** 60

Instructions to Candidates.

Four questions. Attempt **three** questions.

Question 1	SORT, Input & Output Procedures, Control breaks	(20 marks)
Question 2	OO-COBOL, Tables	(20 marks)
Question 3	Business Oriented Programming Language theory	(20 marks)
Question 4	Program Design, Tables, String manipulation, Searching	(20 marks)

Q1 A program is required that will use the CAO Applications File to produce a summary file that shows the number of preferences of each type (first, second, third etc) achieved by UL courses for which preferences are expressed. The summary file must be ordered on ascending Course Code.

The CAO Applications File is a sequential file ordered on ascending CAO-Number. It contains applicant details. The courses that the applicant applies for are held in a 10 element table. The first element of the table contains the first preference, the second the second preference and so on.
Note that the first two characters of the Course-Code represent the institution. All UL courses are prefixed by the LM code.

Field	Type	Length	Occurrence	Value
CAO-Number	X	8	1	-
Applicant-Name	X	60	1	-
Pref-Course-Code	X	5	10	e.g LM051

The Summary File is a sequential file ordered on ascending Course Code. It contains a record for each UL course for which applicants have expressed a preference. Each record in the file has the following description.

Field	Type	Length	Occurrence	Value
Course-Code	X	5	1	e.g LM051
Pref-Count	N	4	10	0-9999

In previous assessment this problem was solved using a table based approach. In this instance you are required to create a program that implements a SORT and control break based solution.

To solve the problem using a SORT and control break based approach you should apply the technique of "beneficial wishful thinking". Imagine how easy the problem would be to solve using control breaks if the input file consisted of records that contained only Course Code and a Preference Indicator (1-10 where 1 is first preference etc) fields and was sorted on these fields. An Input Procedure should be used to create the desired records from the existing input file and an output procedure should be used to generate the summary records.

Write only the Data Division and Procedure Division parts of the program as your answer.
Marks will be apportioned as follows;

a) Applications File, Summary File, and Sort File descriptions (4 marks)
b) SORT with Input Procedure (8 marks)
c) Output Procedure to Generate the Summary Records (8 marks)

Q2. An OO-COBOL class called *"zodiac"* has been written. The class provides a number of methods but only two - "getSignHouse" and "getSignName" - are of interest for this question. The outlines for these methods, showing the parameters they take, are given below.

Using the Zodiac class write a program to process the Students File, count the number of male students and female students born in each house of the Zodiac (Aries, Taurus, Gemini etc), and then display the accumulated results. An example of the expected output is shown below.

The Students file is a sequential file held in ascending StudentId order. Each record of the Students file contains the following items;

FIELD	TYPE	LENGTH	VALUE
Student Id	9	7	0-9999999
Student Name	X		Group
Surname	X	18	-
Initials	X	2	-
DateOfBirth	X	8	DDMMYYYY
Gender	X	1	M/F

Example Run

```
House Sign         Males Females
  1   Aries           50     34
  2   Taurus         150    123
  3   Gemini          75     89
  4   Cancer          94    127
  5   Leo             23     23
  6   Virgo           45     34
  7   Libra           33     46
  8   Scorpio         23     67
  9   Sagittarius     56     60
 10   Capricorn       88     93
 11   Aquarius       102     87
 12   Pisces          44     74
```

Marks for the solution will be allocated as follows –

a) OO declarations (2 marks)
b) File & Record declarations (2 marks)
c) Table declarations (2 marks)
d) Calling methods (2 marks)
e) Accumulate results (6 marks)
f) Format & display accumulated results (6 marks)

```
METHOD-ID. "getSignHouse".
LINKAGE SECTION.
01 InDate.
   02 InDay       PIC XX.
   02 InMonth     PIC XX.
01 ZodiacHouse    PIC 99.
01 OpStatus       PIC 9.
* value of 0 indicates operation was successful
PROCEDURE DIVISION USING InDate, ZodiacHouse RETURNING OpStatus.
*Accepts a date in form DDMM and returns the Zodiac House value (1-12)
END METHOD "getSignHouse".

METHOD-ID. "getSignName".
LINKAGE SECTION.
01 ZodiacHouse    PIC 99.
01 OutSignName    PIC X(11).
01 OpStatus       PIC 9.
* value of 0 indicates operation was successful
PROCEDURE DIVISION USING ZodiacHouse, OutSignName RETURNING OpStatus.
*Accepts a Zodiac House value and returns the Zodiac Sign name
END METHOD "getSignName".
```

Q3 COBOL theory

(a) Briefly outline why, given its target problem domain, COBOL's native support for Decimal Arithmetic is regarded as one of its most important features. (6 marks)

(b) You have written a sub-program which reads a relative file to return the appropriate record for a given key value. To avoid complications you have used the IS INITIAL phrase and have chosen to open the file, read the record, and close the file each time the sub-program is called. Your manager has objected to this approach. Briefly outline what the basis of his objection might be. (4 marks)

(c) Records in an Indexed file are physically organized on ascending Primary Key. Briefly explain how it is possible to read an Indexed file sequentially on one of its alternate keys. (3 marks)

(d) COBOL subprograms have characteristics that limit their effectiveness in creating abstract data types. Briefly, and with the use of some small examples, describe these characteristics. (3 marks)

(e) Define the terms "vertical software market" and "horizontal software market" and briefly describe the characteristic differences between the applications that operate in each of these software markets. (4 marks)

Q4 Microsoft gets customer emails reporting bugs in their products. These emails are collected together into comma separated sequential file. Each record in the file contains a single email and has the following format – *SenderName*, *EmailMessage*. For convenience, assume that the sender name is no more than 40 characters and that the email is no more than 200 characters.

A program is required that will search the emails for the keywords – Excel, Money, Powerpoint, Streets, Windows 2000, Word, XP – and will display on the screen the name of the sender, the keyword that occurs most frequently (dominant keyword) in the email, and the number of times it occurs. The case of the keyword is not relevant. For instance, XP, xP, xp, and Xp are all occurrences of the same keyword and must be counted. In addition, keywords followed by the punctuation marks colon, semi-colon, full-stop, and the question mark must also be counted.

A Program Structure Diagram representing a solution to the problem is given on the next page. Complete the diagram by assigning the Executable Operations shown on this page and by creating and assigning the iteration and selection/search conditions. Sign the completed diagram, detach it from this exam paper, and enclose it in your answer book.

Marks for the solution will be allocated as follows –

a) Assign Executable Operations correctly (10 marks)
b) Create appropriate iteration and selection/search conditions (6 marks)
c) Assign the iteration and selection/search conditions correctly (4 marks)

Some Data Items

```
01 DetailLine.
   02 Prn-SenderName  PIC X(40).
   02 Prn-DomKeyword  PIC BBX(12).
   02 Prn-DomOccurs   PIC BBZZ9.

01 KeywordTable.
   02 TableValues.
      03 FILLER       PIC X(11) VALUE "EXCEL      ".
      03 FILLER       PIC X(11) VALUE "MONEY      ".
      03 FILLER       PIC X(11) VALUE "POWERPOINT ".
      03 FILLER       PIC X(11) VALUE "STREETS    ".
      03 FILLER       PIC X(11) VALUE "WINDOWS2000".
      03 FILLER       PIC X(11) VALUE "WORD       ".
      03 FILLER       PIC X(11) VALUE "XP         ".
   02 FILLER REDEFINES TableValues.
      03 Keyword      PIC X(11) OCCURS 7 TIMES INDEXED BY Kidx.

01 KeywordOccurences.
   02 OccurCount      PIC 999   OCCURS 7 TIMES INDEXED BY Oidx
```

Executable Operations

```
1.   DISPLAY HeadingLine
2.   DISPLAY DetailLine
3.   MOVE SenderName TO Prn-SenderName
4.   MOVE DomKeyword TO Prn-DomKeyword
5.   MOVE DomOccurs  TO Prn-DomOccurs
6.   MOVE OccurCount(Oidx) TO DomOccurs
7.   MOVE Keyword(Kidx)    TO DomKeyword
8.   MOVE OccurCount(1) TO DomOccurs
9.   MOVE Keyword(1)    TO DomKeyword
10.  SET Oidx TO Kidx
11.  ADD 1 TO OccurCount(Oidx)
12.  SET Kidx TO 1
13.  UNSTRING Email DELIMITED BY ALL SPACES INTO CheckWord
              WITH POINTER UnstringPtr
14.  MOVE 1 TO UnstringPtr
15.  MOVE ZEROS TO KeywordOccurences
16.  MOVE FUNCTION UPPER-CASE(Email) TO Email
17.  INSPECT Email CONVERTING ".;:?" TO SPACES
18.  INSPECT Email REPLACING ALL "WINDOWS 2000" BY "WINDOWS2000 "
19.  UNSTRING EmailRec DELIMITED BY ","
     INTO SenderName, Email
20.  READ EmailFile AT END SET EndOfEmailFile TO TRUE END-READ
21.  OPEN INPUT EmailFile
22.  CLOSE EmailFile
```