UNIVERSITY *of* LIMERICK

O L L S C O I L   L U I M N I G H

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

Assessment Paper

Academic Year: 2007/2008   **Semester :** 2
Module Title: Leveraging Legacy Applications   **Module Code:** CS4558
Duration of Exam: 2.5 hours   ***% of Total Marks:*** 100%
Lecturer: Michael Coughlan   **Paper marked out of:** 150

Instructions to Candidates.

4 questions. Attempt 3. All questions carry equal weight.

Question 1   Legacy Systems (general)   (50 marks)

Question 2   Code Renovation (practice)   (50 marks)

Question 3   Data Migration   (50 marks)

Question 4   Code Renovation and Language Conversion   (50 marks)

Q1. Legacy Systems - General

(a) Studies have identified four different types of maintenance: corrective, adaptive, perfective and preventive. Classify each of the following maintenance activities in this way and provide a justification for your classification:

   i. Rewrite a complex method, replacing a series of goto's and conditional statements with PERFORM loops.

   ii. Address a user complaint that the print function on the View Customers screen no longer seems to work.

   iii. Increase the size of a table to allow a program to correctly process longer lists of items than are currently allowed.

   iv. Make modifications to the invoicing subsystem to allow invoices to be generated in Euros, rather than Pounds Sterling, for selected customers.

(10 marks)

(b) Legacy systems are said to be difficult, expensive, and time consuming to maintain. Identify and briefly describe the factors that make maintenance of legacy systems so problematic.

(10 marks)

(c) Using examples taken from case studies and guest lectures identify and describe four factors that might prompt Legacy System Modernisation. (8 marks)

(d) Replacement, Retargeting, Revamping, and Code Renovation are all approaches that might be considered when a legacy system needs to be modernized. Briefly describe what is involved in each of these solutions and comment on the cost, difficulty, and effectiveness, of each. (12 marks)

(e) Legacy system data is often stored in flat files, structured files, or non-relational databases. At some time in a system's life cycle modernization efforts may require the data to be migrated to a modern database. Outline, and briefly describe, the common risks that such a data migration entails. Support your answer with some examples drawn from the experiences recounted by guest lecturers. (10 marks)

Q2. Code Renovation (practice)

(a) Legacy systems are difficult to maintain and difficult to enhance. Yet despite these difficulties revitalizing a legacy system is often seen as a more desirable alternative to redeveloping it. Briefly explain why renovation is so often seen as a preferable alternative to redevelopment. (8 marks)

(b) Examine the program fragment below (next page) and -

   i. For each paragraph in the program state whether or not it is an *internal-procedure-like* paragraph and give the reasons why it is, or is not, such a paragraph. (12 marks)

   ii. For a paragraph you have nominated as an *internal-procedure-like* paragraph, identify its Localizable and Pseudo Localizable variables, state which is which, and state the criteria by which you so identify these variables. (14 marks)

   iii. Show, by producing a skeleton of the program fragment highlighting the required changes, how the *internal-procedure-like* paragraph you selected can be converted into a parameterized Contained Subprogram. State the criteria used to identify the parameterizable variable(s) and the parameter passing mechanism(s). (16 marks)

```
       DATA DIVISION.
       FILE SECTION.
       FD  StudentFile-CSV.
       01  StudentRec-SF        PIC X(100).

       WORKING-STORAGE SECTION.
       77 StudentName           PIC X(30).
       77 StudentAddress        PIC X(70).
       77 CountyNum             PIC 99.

       01 UnstringPointer        PIC 99.
          88 EndOfAddress        VALUE 71.

       01 CIdx                   PIC 99.
          88 NoValidCounty       VALUE ZEROS.

       01 DisplayLine.
          02 CountyNamePrn           PIC X(9).
          02 FILLER                  PIC X(6) VALUE SPACES.
          02 CountyTotalPrn          PIC ZZ,ZZ9.

       01 CountyCount OCCURS 26 TIMES PIC 9(5).

       01 CountyNameTable.
          02 TableValues.
             03 FILLER  PIC X(27)  VALUE "Carlow    Cavan     Clare".
             03 FILLER  PIC X(27)  VALUE "Cork      Donegal  Dublin".
             03 FILLER  PIC X(27)  VALUE "Galway    Kerry     Kildare".
             03 FILLER  PIC X(27)  VALUE "Kilkenny Laois     Leitrim".
             03 FILLER  PIC X(27)  VALUE "Limerick Longford Louth".
             03 FILLER  PIC X(27)  VALUE "Mayo      Meath     Monaghan".
             03 FILLER  PIC X(27)  VALUE "Offaly    RoscommonSligo".
             03 FILLER  PIC X(27)  VALUE "TipperaryWaterfordWestmeath ".
             03 FILLER  PIC X(18)  VALUE "Wexford   Wicklow".
          02 FILLER REDEFINES TableValues.
             03  CountyName  PIC X(9) OCCURS 26 TIMES
                                      INDEXED BY NameIdx.
       PROCEDURE DIVISION.
       Begin.
          OPEN INPUT StudentFile-CSV
          PERFORM CountCounties.

       DisplayResults.
          DISPLAY "Student County Totals"
          DISPLAY "CountyName        CountyTotal"
          PERFORM VARYING NameIdx FROM 1 BY 1 UNTIL NameIdx > 26
             SET CIdx TO NameIdx
             MOVE CountyName(NameIdx)    TO CountyNamePrn
             MOVE CountyCount(CIdx)      TO CountyTotalPrn
             DISPLAY DisplayLine
          END-PERFORM
          CLOSE StudentFile-CSV
          STOP RUN.

       GetStudentCountyNum.
          MOVE 1 TO UnstringPointer.
          UNSTRING StudentAddress DELIMITED BY ALL SPACES
              INTO CountyNamePrn
              WITH POINTER UnstringPointer.
          PERFORM UNTIL EndOfAddress
             UNSTRING StudentAddress DELIMITED BY ALL SPACES
                 INTO CountyNamePrn
                 WITH POINTER UnstringPointer
          END-PERFORM
          SET NameIdx TO 1
          SEARCH CountyName
              AT END MOVE ZEROES TO CountyNum
              WHEN CountyName(NameIdx) = CountyNamePrn
                  SET CountyNum TO NameIdx
          END-SEARCH.

       CountCounties.
          READ StudentFile-CSV
              AT END GO TO DisplayResults
          END-READ
          UNSTRING StudentRec-SF DELIMITED BY ","
                   INTO StudentName, StudentAddress
          PERFORM GetStudentCountyNum
          MOVE CountyNum TO CIdx
          IF NoValidCounty
              DISPLAY "No valid county found in address"
            ELSE
              ADD 1 TO CountyCount(CIdx)
          END-IF
          GO TO CountCounties.
```

Q3. Data Migration

(a) One way to reduce the risks of data migration is through Data Cleaning.   Briefly describe what is meant by the term Data Cleaning and outline why it might be required as part of a data migration initiative.   .   (8 marks)

(b) Briefly describe, and give examples of, some of the data quality problems to be found at the schema-level and instance-level of single-source legacy data.   (12 marks)

(c) A Data Redundancy Consolidation initiative involves combining existing physical structures (data stores) into a single structure. Identify, and describe, the main schema-level and instance-level data quality problems that might have to be addressed as part of this initiative.
                                                                                        (10 marks)

(d) The *Composite Gateway* approach to data migration may be regarded as a combination of the *Database First* and *Database Last* approaches. Briefly, and with the aid of diagrams, explain how these two approaches work.   (8 marks)

(e) The Butterfly Methodology is data migration approach developed in Trinity College as part of the MILESTONE project. Outline the 6 phases of data migration using the Butterfly Methodology and describe how this approach to data migration works.   (12 marks)

Q4. Code Renovation and Language Conversion

(a) Veerman, in his paper "Revitalizing Modifiability of Legacy Assets", describes a number of restructuring transformations that might be applied to legacy code. Identify, and briefly describe, the purpose of the three types of transformations outlined by Veerman and mention a number of the transformation activities associated with each type.   (10 marks)

(b) Both the "Realities of Language Conversions" and "Revitalizing Modifiability of Legacy Assets" have example COBOL code that makes use of a "Bar SECTION". Briefly describe the purpose of the "Bar SECTION", show how it aids program maintenance, and outline its role in the program restructuring transformations.   (10 marks)

(c) What is "data type emulation" and why would it be necessary if we were converting a COBOL application to a language like C++?   (12 marks)

(d) What drawbacks/problems might be experienced with data type emulation?   (8 marks)

(e) Making a legacy system more maintainable, change-enabled, contemporary, or component based, are often reasons given for attempting a language conversion. Briefly comment on how effectively a language conversion can deliver on these objectives.   (10 marks)