University of Swaziland

Department Of Computer Science

Supplementary Examination

JULY 2013

Title of paper :	*C under Unix*
Course number :	*CS344*
Time Allowed :	*Three(3) hours*
Instructions :	

- *Each question is worth 25 marks*
- *Answer question 1*
- *Answer any three(3) questions from questions 2 to 6*

This paper may not be opened until permission has been granted by the invigilator

Question 1-*compulsory*

(i) Explain the following concepts. Give Examples where possible?

(a) Function Prototype *1 mark*

(b) Function Signature *1 mark*

(c) Function Overloading *1 mark*

(d) Friend function *1 mark*

(e) Access member function *1 mark*

(ii)

(a) What is a constructor function? *2 marks*

(b) How and when is a constructor function called/used ? *2 marks*

(c) A class may have more than one constructor function. How is this possible? *2 marks*

(iii)

(a) What is a polymorphic function? *2 mark*

(b) Explain two (2) cases (concepts) which give rise to polymorphic functions. *2 marks*

(c) What is function template? *2 marks*

(d) Do you see any relationship between a function templates and polymorphism? *2 marks*

(iv)

a) What is the relationship between a class and an object? *2 marks*

b) What is difference between private and protected members of a class? *2 marks*

c) What is the difference between a private and a public base class? *2 marks*

Question 2

(i) Assume that the charge for sending a parcel overseas is calculated as follows. First, its weight is rounded up to the nearest multiple of 15 grams. Then the charge is computed using the following table;

Weight (*grams*)	**Charge** (*cents*)
15	12
30	22
45	31
60	36 plus 2 per each completed 1000 km
75 and over	40 plus 3 per each completed 1000 km

Write a C++ function **ComputeCharge** that takes the *Weight* of the parcel and the *Distance*, as arguments, and returns the Charge in cents. *10 marks*

(ii) Write a C++ function **DoubleSpace** *(ifstream InFile, ofstream OutFile)* that takes an input stream, *InFile*, as input and copies all the contents of InFile to an output stream, *OutFile*. The lines in OutFile must be double spaced. That is there must be at least one blank line between any consecutive lines in OutFile. The figure below shows a sample InFile and the expected double spaced OutFile.

InFile – *Single Spaced*

```
Good Hamlet,
Cast thy nited color
And look like a friend to Denmark
```

OutFile – *double spaced*

```
Good Hamlet,

Cast thy nited color

And look like a friend to Denmark
```

15 marks

Question 3

Consider rational numbers (fractions) of the form: **a/b**, where **a** is the numerator and **b** is the denominator. The arithmetic operations addition, subtraction, multiplication and division can be performed on rational numbers. These operations can be performed as shown in the following examples.

a/b + c/d = (ad + bc) / bd
a/b - c/d = (ad - bc) / bd
a/b * c/d = ac/bd
(a/b) / (c/d) = ad/bc

(i) Using C++ notation, define class ***Rational*** for performing arithmetic operations with fractions. Use integer variables to represent the numerator and the denominator of a fraction. In the definition of the class provide prototypes for member functions to perform the arithmetic operations addition, subtraction, multiplication and division. Use operators instead of functions as much as possible. For example use operator + instead of function name **add**. The class should also specify a member function (or operator) for printing a rational number in the form **a/b,** where **a** is the numerator and **b** is the denominator. Your class should have two (2) constructor functions. The first constructor should have no arguments and simply initializes the numerator and denominator to be 0 and 1 respectively. The second constructor takes two (2) integer arguments and initializes the numerator and denominator to be the first and second arguments respectively. *10 marks*

(ii) Using C++ notation, write the code (definition) of the member functions whose prototypes are specified in the definition of class ***Rational*** above.

15 marks

Question 4

(i) Distinguish between single inheritance and multiple inheritance. *2 marks*

(ii) Using the concepts of classes, subclasses and inheritance, model (use diagram) an account class hierarchy for a banking enterprise. For each account the account number, name of holder (owner) and balance are recorded. Account holders may deposit or withdraw money from the account. A deposit transaction increments the balance by the amount being deposited and withdrawal decrements the balance by the amount being withdrawn.The enterprise has two types of account: A current (cheque) account and a savings account. For each current account the overdraft limit is recorded. The overdraft limit is the amount by which the account holder is allowed to overspend each month. The overdraft limit may be changed from time to time. Each savings account attracts an annual interest rate. This rate may be changed from time to time. The interest for each savings account is calculated each month and deposited into the account. The following formula may be used for this purpose: MonthlyInterest = Annualrate * balance / 12. The details for each account may be printed when necessary. For a current account the account number, name of owner, balance and overdraft limit may be printed. For a savings account the account number, name of owner, balance and interest rate are printed.

10 marks

(iii) Using C++ notation write an implementation of the class hierarchy obtained above. That is define all the classes and write the corresponding code for the member functions in each class.

13 marks

Question 5

Write a program that implements an array, L [0..N] of linked lists. The program must read a sequence of integers from a text file. For each integer X read from the text file the program must insert X into linked list L[I] only if X is a multiple of I. Once the program has read all the integers from the text file it must display all the values in each linked list on standard output.

Assume class **LinkedList** is already defined with the operations

- *LinkedList()* – constructor for class linked list and initializes list.
- *Insert (e)* – Insets new element e into linked list. For example S.Insert(10) inserts the value 10 into linked list S.
- *Delete (e)* – Delete element e from the linked list
- *Show()* – Displays all the elements in the linked list. . For example S.Show() displays all the element in linked list S on standard output. 25 *marks*

Question 6

(i) Using a template, write a C++ *Single Un-Ordered Linked List* class definition.

6 marks

(ii) Using your class definition in (i) above, write the suitable function definitions for the ***List*** constructor and the member functions ***Insert, Delete and IsEmpty***.

12 marks..

(iii) Show how your Linked List class definition may be used to declare a List of integer numbers or real numbers or characters. *3 marks*

(iv) What would you need to change in the above definition if you wanted your list to be *Ordered* instead of being *Un-ordered.* *4 marks*