University of Swaziland
Final Examination
MAY 2009

Title of paper : Data structures

Course number : CS342

Time Allowed : Three(3) hours

Instructions :

- ***Each question carries 25 marks***
- ***Answer any four (4) questions from questions 1 to 6.***

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

Question 1

(a) What is the meaning of the following terms/phrases *5 marks*

(i) abstraction.
(ii) the running time of an algorithm
(iii) recursive algorithm
(iv) An logarithmic algorithm
(v) base address of a record.

(b) Consider the following record declaration.

```
Var  P : Record
                PIN          : string[5]
                Firstname    : sting[10]
                Lastname     : string[30]
                Age          : Integer;
             End;
```

(i) Assume each character requires *1 byte*, each integer requires *4 bytes* and the base address of P is 500. A string is an array of char. Let P have the following values.

P

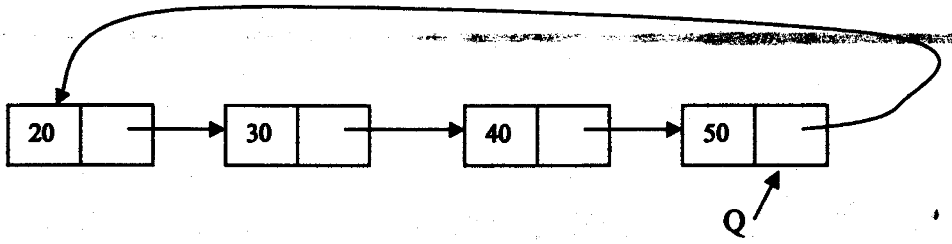
PIN	**Firstname**	**Lastname**	**Age**
191818	james	king	44

(a) Show the allocation of record P in RAM. *5 marks*
(b) How many bytes are used to store record P. 4 *marks*
(c) What is the base address of each field of P. 6 *marks*

(ii) Let L be a list of N records similar to P above. What is the total number of bytes required to store L assuming a linked list implementation of L. *5 marks*

Question 2

A circular queue may be implemented using pointers as shown in the following diagram.



The leftmost node is in the front of the queue, and the rightmost node is in the back of the queue. Pointer Q points to the back of the queue. Elements are added to the back of the queue and removed from the front of the queue. Using C++ (or Pascal) notation, describe how the queue may be implemented. You must define the structure (using C++ or Pascal notation) and the operations to add and remove elements to/from the queue. For each operation, write the code and estimate its running time using big-oh notation. *25 marks*

Question 3

(a) Construct a **binary search tree** T containing the following values.
100 98 10 20 40 2 25 15 70 75 65 200
150 30 170

Assume values are inserted in the given order. *7 marks*

(b) What is the root node of tree T constructed above? *1 mark*

(c) What is the height of tree T ? *1 mark*

(d) What is the depth of node 200 in tree T? *1 mark*

(e) What is the path from the root node to node 30? *1 mark*

(f) List all node values in left shell of the root node of T? *3 marks*

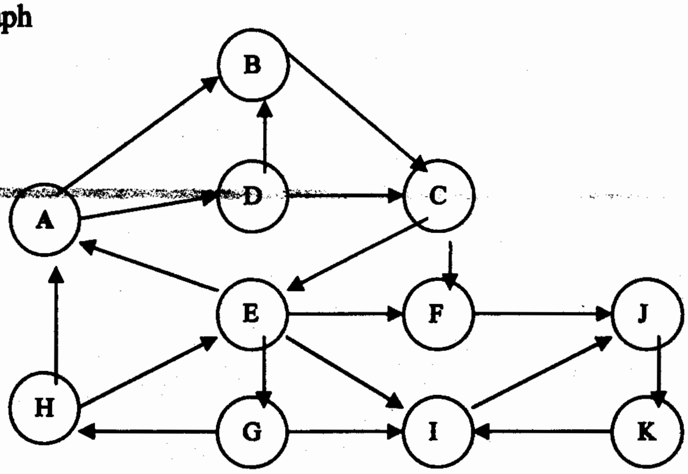
(g) List all node values in T assuming post order traversal? *3 marks*

(h) Trace the execution of iterative level order traversal algorithm on tree T.

8 marks

Question 4

Consider the following graph



(a) Show the adjacency matrix representation of the above graph G. *4 marks*

(b) Show the adjacency list representation of the above graph G. *4 marks*

(c) List all the nodes of G, assuming Breadth-First Search (BFS) starting from node E. You may assume adjacent nodes are visited in alphabetical order. *7 marks*

(d) The Depth-First Search (DFS) algorithm can be implemented using as stack as follows:

```
DFS ( v, G)  -- depth-first search graph G starting from node v.
Begin
  Push node v onto stack S.
  Process and mark node v as visited.

  While stack S is not empty do
  Begin
    Let w be the node on top of stack S
    If all nodes adjacent to w have been visited then
        Pop node w from stack S.
    Else
        Begin
          Select an unvisited node u adjacent to w.
          Push node u onto stack S.
          Process and mark node u as visited.
        End;
  End;
End;
```

Trace the execution of the Depth-First Search (DFS) algorithm on the above graph G starting from node E. Assume adjacent nodes are visited in alphabetic order. *10 marks*

Question 5

(a) Using the C++ standard template library;

(i) Define a class **GraphNode** to represent a node in a graph with *NodeValue* and *AdjacentList* as its data members. The node value is of type *datatype and AdjacentList* is a list of values of type *datatype*. [***do not define any memberfunction s for this class***] *4 marks*

(ii) Using your class definition in (i) above, define a class **Graph** with *Nodes* as its data member where *Nodes* is a vector of GraphNodes. [***do not define any memberfunction s for this class***] *2 marks*

(b) Assuming the type definition in (a.) above, write C++ code for the following member functions of class Graph.

(i) NodeIndex (u) – returns the index of node u in a graph. *4 marks*

(ii) IsNode (u) – returns true if u is a node in the graph and returns false otherwise. *4 marks*

(iii) AddEdge (u,v,) – adds a new edge from node u to node v. *4 marks*

(iv) Neighbours (u) – displays all the neighbors of node u (the adjacent list of node u) in a graph. *4 marks*

(c) Estimate the running time of each of the operations implemented in (b.) above. *3 marks*

Question 6

(a) Write the pseudocode for inserting into a B-tree of order b. *4 marks*

(b) Follow the pseudocode outlined in (a) above and construct a B-tree of order 5 containing the following values.

100 50 10 20 40 60 25 15 70 75 65 200
150 30 170 38 35 39

Assume values are inserted in the given order. *15 marks*

(c) List all node values in the B-tree constructed above assuming pre-order traversal. What is the running time of this traversal? *3 marks*

(d) What is the height of a B-tree of order 19 containing 130000 values. *3 Marks*