THE OPEN UNIVERSITY OF SRI LANKA
B.Sc DEGREE PROGRAMME: LEVEL 03
FINAL EXAMINATION: 2009/2010
CPU1142: DATA STRUCTURES AND ALGORITHMS

DURATION: **TWO HOURS** (2 HOURS)

090

Date: 20th July, 2010 Time: 9.30 am – 11.30 am

Answer FOUR Questions ONLY.

Q1.

a) Define the term *Abstract Data Type (ADT)*.

b)

i. Explain the solutions to overcome the main problems caused, when selecting a data structure.

ii. List out the areas in which data structures are applied.

c) Describe the *array implementation* and the *pointer implementation* of the linked list data structure.

d)

	Info	Next
0	5	10
1	10	14
2		
3	45	9
4	33	-1
5	14	6
6	20	12
7		
8		
9	22	5
10	12	3
11		
12	50	1
13		
14	15	4

The above array illustrates an *array implementation* of a linked list. Considering that, write down the order of the values which are stored in the list starting from index 0.

e) Consider the following structure of a *Pointer implementation of list* and answer the questions from (i) to (iii).

```
struct node
{
   int info;
   struct node *ptr;
}
Typedef struct node nodePtr;
```

Write C program code to do the following:

i. Insert a node at the beginning of the list.
ii. Print the list.
iii. Delete a node at the end of the list.

Q2.

a)

i. Distinguish between the *Linear linked list* and *Circular linked list*.
ii. What are the disadvantages of the circular linked list?

b) What is a *queue*?

c) Explain the reason why the circular queues are used rather than the linear queues in some instances.

d) Circular queue has the following basic operations.

```
void insert(struct queue *p, int item)
```
Adding a new item to the rear of the queue.

```
int remove(struct queue *p)
```
Remove the front item from the queue and returning it.

i. Write an algorithm to perform the `insert` function.

ii. Following array shows the current state of a circular queue. Graphically show the following operations that can be performed on the queue by indicating the *front* and *rear* pointers.

```
insert (Q, 40);
insert (Q, 12);
insert (Q, 50);
insert (Q, 25);
remove (Q);
```

5	
4	
3	10
2	9
1	
0	

Q3.

a) Construct a binary tree by considering the following traversals of a tree.

Preorder : A B C E D F G H J I
Inorder : E C D B F G A J H I

b) Answer the following questions from (i) to (iv) by using the constructed binary tree in part (a).

i. Write the postorder traversal of the tree.
ii. What is the depth of the tree?
iii. What are the leaf nodes and non leaf nodes?

iv. Is the tree an almost complete binary tree? If not state the rule that has been violated.

c) Construct binary trees of the following mathematical expressions and write down the Prefix, Infix and Postfix forms of each expression.

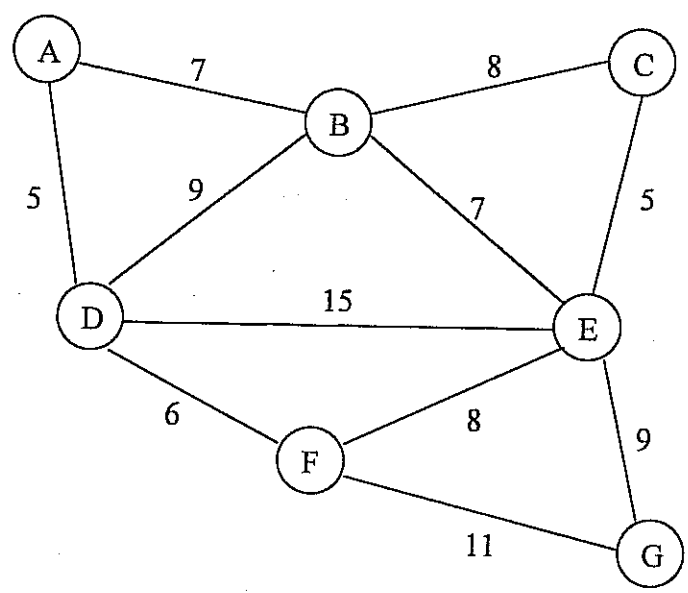
i. $A + B - C$
ii. $(A + B) * (C - D)$
iii. $(A + B * C) \$ ((A + B) * C)$

Q4.

a)

i. What is a *Graph*?
ii. Draw a weighted directed graph that has a cycle.
(Note: It should have more than 6 vertices.)

b) Consider the following graph and answer the questions.



i. Represent the above graph as an adjacency matrix and an adjacency list.
ii. Using the adjacency representations, apply the depth-first search to the graph and list the vertices they would be visited starting from node A.
iii. Draw the associate minimum spanning tree for the above graph.

Q5.

a) Distinguish between the *Internal sorting methods* and *External sorting methods*.

b) Write C program code to sort a set of integers using the *insertion sort* method.

c) Explain how your program works on the following data set.

18 15 7 2 20 3 5 14

d) Briefly describe the shell sorting algorithm by using an example.

e) A heap is a binary tree. Write two rules that the binary tree must follow in order for the structure to actually be a heap.

f) Draw an initial max-heap for the following set of integers.

25 57 48 37 12 92

Q6.

a) What is meant by the *Sequential search*? Explain using an example.

b)

i. Construct a binary search tree for the following set of integers.

12, 24, 6, 14, 2, 18, 9, 12, 7, 36, 24, 18, 5

ii. Explain how to find the minimum and maximum keys in a binary search tree.

iii. What will be the result if the node with value 24 is removed from the above constructed binary search tree in part b (i)? Clearly draw the resulting diagram of the above tree structure.

c)

i. What is a *Hash table*?

ii. What is meant by *hash collision*? Describe how to minimize such collisions.