Uva Wellassa University, Sri Lanka
Faculty of Science & Technology
2nd Semester Examination – September/October 2012
CST225-2 Data Structures & Algorithms

Total **05** Questions Time: **Two (02)** hours Answer **all** questions

Question 1. **(20 marks)**

(a) List out ***six*** areas in which data structures are applied extensively.

(b) What are the desirable ***properties*** of an Algorithm? State the purpose of analyzing the complexity of an algorithm.

(c) Assuming the RAM model of computation, analyze the ***running time*** of the following program fragment. Clearly show the steps in each statement.

```
sum ← 10
for i←1 to n - 2
        sum ← sum + A[i]
```

(d) Find the ***Big O*** value for the following functions:

i) $T(n)= 3 +5n + 3n^2$
ii) $f(n)= 2^n + n^2 +8n +7$
iii) $T(n)= n + \log n +6$
iv) $f(n)= 2n^3+n!+12n \log n$

(e) State the ***advantages*** and ***disadvantages*** of following data structures.

i) Array ii) Ordered Array iii) Stack iv) Queue v) Linked List

Question 2. **(20 marks)**

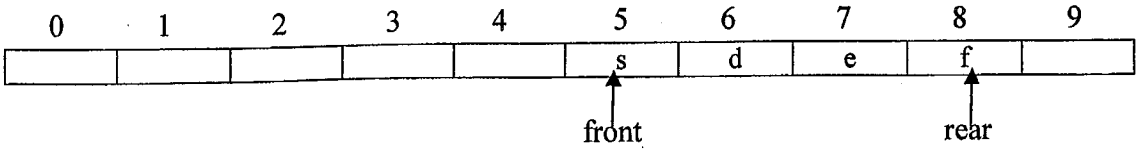
(a) Write the algorithms for following ***stack*** operations;

i) Inserting an item into a stack
ii) Deleting an item from a stack

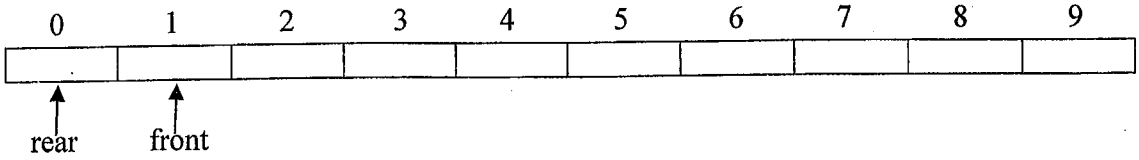
(b) Which data structure is used to perform ***recursion***? Briefly explain.

(c) How to overcome the problem in using "***linear queues***" by using "***circular queues***"?

(d) You are given the following ***circular queue***.



Apply permissible operation on that queue and convert it to the queue given below;



(e) Write the algorithms for following ***queue*** operations;

i) Inserting an item into a queue
ii) Deleting an item from a queue

Question 3. **(20 marks)**

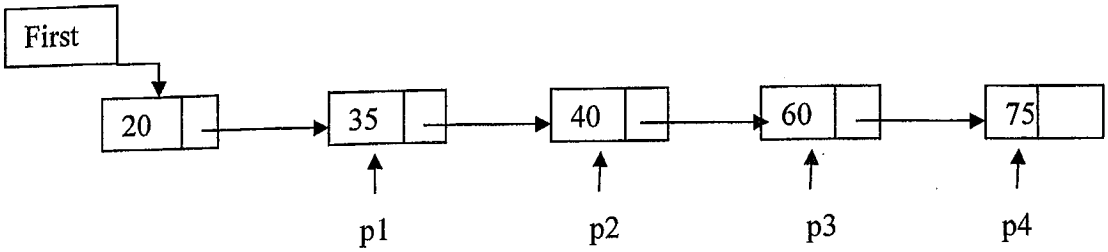
(a) Explain whether linked list is actually ***linear*** or ***non-linear*** data structure.

(b) The given figure has two linked-lists (first column stores data part and second column stores next part) stored in the 2D array. Based on the figure, answer the following questions:

	node	data	next
	[0]	66	-1
	[1]	25	-1
	[2]	9	0
First_list_start --->	[3]	33	8
	[4]	11	1
Second_list_start--->	[5]	10	9
	[6]	7	2
	[7]	18	4
	[8]	21	6
	[9]	48	7

i) If we run ***traverse()*** on the First_list, what is the output suppose to be?
ii) If we run ***traverse()*** on the Second_list, what is the output suppose to be?

(c) You are given the following linked list of link objects and references ***p1, p2, p3, and p4.*** For each code segment, draw a similar figure indicating how the list changes.



i) p2 = p1.next;
ii) first = p1.next;
iii) p3.data = p1.data;
iv) p3.data = p1.next.data;
v) p1.next.data = p1.data;

(d) Give an algorithm to ***reverse*** the elements of a single linked list without using a temporary list.

(e) Write the algorithms to search a given element from a data array using following two searching methods and comment on the complexity.

i) Linear search
ii) Binary search

Question 4. **(20 marks)**

(a) Following is an algorithm for ***insertion sort*** which takes an array 'A' as input.

```
INSERTION-SORT (A)
1     for j ← 2 to length [A]
2          do key ← A [j]
3               insert A[j] into the sorted sequence A [1..j-1]
4               i ← j-1
5               while i > 0 and A [i] > key
6                  do A [i+1] ← A [i]
7                       i ← i – 1
8                  A [i+1] ← key
```

Illustrate the operation of *INSERTION-SORT* on the array ***A = {6, 3, 5, 7, 2, 4}***

(b) What is ***Divide and Conquer*** method? What are the advantages of it?

(c) What are the properties of the ***pivot value*** in quick sort? How do you select the pivot value in your algorithm?

(d) Apply ***quick sort*** on following array. (Show all the steps)
{4, 8, 1, 6, 3, 7, 2, 5}

(e) Write the algorithm for ***merge sort.*** Apply ***merge sort*** on following array. (Show all the steps)

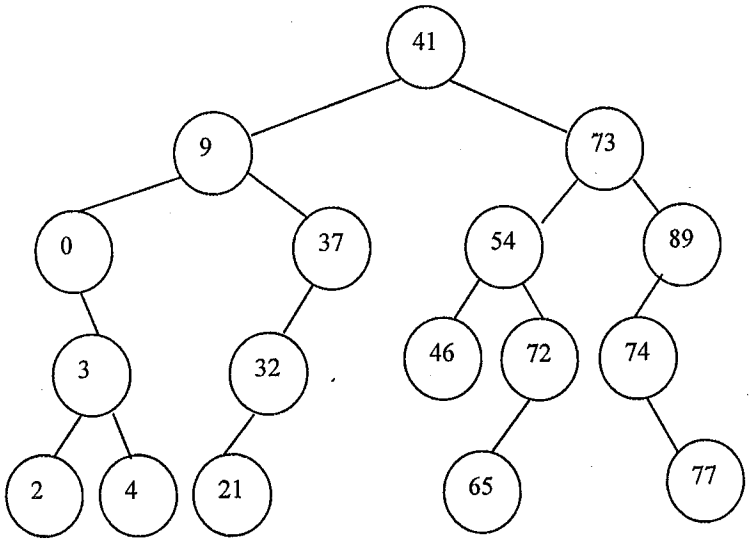
0	1	2	3	4	5	6	7	8	9	10	11
64	21	33	70	12	85	44	3	97	24	51	40

Question 5. **(20 marks)**

(a) What are the ***three*** types of binary trees available? Give ***one*** condition that each tree should satisfy.

(b) Draw a binary tree for the following expression;
A*B-(C+D)*(P/Q)

(c) Consider the following binary search tree;



What is the sequence of numbers that would result, when traversed as;

i) In-order ii) Pre-order iii) Post-order

(d) Give the iterative algorithm for the ***in-order*** traversal of a binary tree.

(e) Write the algorithms to following ***deletion operations*** in a binary tree;

i) Node to be deleted has one child.
ii) Node to be deleted has two children.

---End of Question Paper---