6

00060

HE OPEN UNIVERSITY OF SRI LANKA
.Sc. DEGREE PROGRAMME: LEVEL 04
EPARTMENT OF MATHEMATICS AND COMPUTER SCIENCE
INAL EXAMINATION – 2011/2012
CSU2279: DATA STRUCTURES AND ALGORITHMS
URATION: THREE HOURS (3 HOURS)

ate: 24th November 2012 **Time: 1.30pm - 4.30pm**

nswer Four Questions ONLY.

01.

(a). Discuss why it is important to analyze an algorithm before it is implemented for a particular situation.

(b). Explain the factors on which the running time of a program depends.

(c). Explain the following terms with examples.

(i). Best case running time.

(ii). Average case running time.

(iii). Worst case running time.

(d). Perform a Big-O analysis for each of the following functions.

(i). $\frac{20(n+1)(n-2)^2}{(n^2-4)}$

(ii). $3n^2 + 2n - 1$

(e). Calculate the running time of the following functions.

(i).
```
function fac (i:integer) : integer;
          var
          x : integer;
     begin
          fac := 1;
          for x := 1 to i do
          fac := fac*i;
     end;
```

(ii). Procedure bubble (var A : array[1..n] of integer);

```
        var
              i,j,temp : integer;
        begin
                    for i := 1 to n-1 do
                    for j := n down to i+1 do
                    if A[j-1] > A[j] then

              begin
                    temp := A[j-1];
                    A[j-1] := A[j];
                    A[j] := temp;
              end;
        end;
```

Q2.

(a). What are abstract data types (ADT)? Describe two properties of ADT.

(b).

i). Create an ADT to store marks of mathematics of O/L students. The ADT should co of student's name, index number, and marks obtained. Assume there are 50 student the class.

ii). Write a procedure to display the results sheet in the ascending order of marks of mathemaics marks.

(c). Write a recursive function to print the Fibonacci number series those who are not greater than 50. **For example (1, 1, 2, 3, 5, 8, 13, 21, 34).** The **n**th term of the Fibonacci numb series is given by the following formula.

$\mathbf{F_n = F_{(n-1)} + F_{(n-2)}}$ **(n >= 2).**

$\mathbf{F_0=0}$ **and** $\mathbf{F_1 = 1.}$

(d). Using Quick Sort algorithm show graphically, step by step, how the integers given bel can be sorted.

98, 22, 101, 102, 22, 30, 38, 140, 97, 68

Write short notes on each of the following topics.

i). Balance Multiway merging

ii). Different categories of searching methods.

iii). The straight selection sort algorithm.

State the classification of sorting algorithms. What are the differences between them?

Show how Bubble sort and Straight Selection sort algorithms to sort a set of data in ascending order (given below) work on the following set of integers by dry running. By using your dry running chart, explain any differences between **Bubble sort algorithm** and **Straight Selection Sort algorithm.**

22	12	18	14	11

Bubble Sort algorithm

```
for  i:= 1 to n-1 do
     for  j:=n down to i+1 do;
       if A[j].key<A[j-1].key  then
swap(A[j],A[j-1])
```

Straight Selection Sort algorithms

```
for  i:= 1 to n-1 do
begin
             lowindex := i;
             lowkey := A[i].key;

     for j:= i+1 to n do
     begin
             if A[j].key < lowkey then
             begin
                 lowkey := A[j].key;
                 lowindex := j;
             end;
     end;
       swap(A[i],A[lowindex])
end;
```

Q4.

(a). Compare and contrast the array based implementation and pointer based implementation of the linked list data structure.

(b). State the Advantages and Disadvantages of doubly linked lists.

(c).

(i).Using an appropriate diagram explains the need of **Header** and **Nil** pointers in ***singly linked lists.*** Is there any difference between them?

(ii).Using appropriate diagrams, show how the **Insert** and **Delete** operations of a pointer based ***doubly linked lists*** work.

(d). Convert the following pointer based definition in to its array based definitions

```
type
        celltype = record
                value : valuetype;
                next : ^celltype;
        end;
list : ^celltype;
```

(e). Write a Pascal procedure/function to **INSERT** a node into an array based implementation of a linked list.

(f). Write a Pascal procedure/function to **LOCATE** for a given element in an array based implementation of a linked list.

Q5.

(a). State two instances where the stack data structure can be used.

(b). State the difference between the Queue data structure and Stack data structure by using appropriate diagrams.

(c). Using an appropriate diagram explain the circular array implementation of a Queue.

(d). Write an array based PASCAL procedures/functions for the following stack operations.

i). MAKESTACK(S)	: Make stack S being empty stack.
ii) EMPTY(S)	: Return true if S is an empty stack.
iii) PUSH(X, S)	: Insert the element x at top of stack s.
iv) POP(S)	: Delete the top element of the stack.

Q6.

(a). Briefly explain the following terms with a suitable example using more than six ***(6) nodes*** with respect to a binary tree structure.

(i). Strictly binary

(ii). Complete binary.

(iii). Almost binary.

(b). Represent the following mathamatical expressions using binary trees. Write down the **post order** forms of each expression.

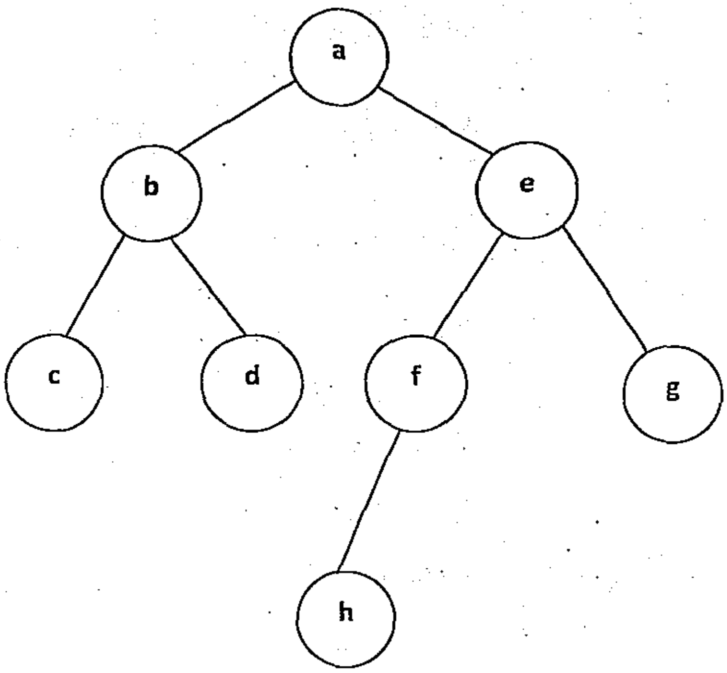
(i) a+b*c+d

(ii). x+y!*z

(iii). (a+b*c) $ ((a+d)*c) **(Where , the $ sign represents the exponention)**

(iv). x*y – (p-q)*(r/s)

(c). Answer the following questions considering the given tree structure **B1.**



B1 Tree

(i) How many levels are there in the tree?

(ii) How many leaf nodes are there?

(iii) Is **B1** a ***Strictly*** binary tree **OR** a ***Complete*** binary tree? Justify your answer.

(iv) Give the **preorder, inorder** and **postorder** traversal of the tree **B1.**

All Right Reserved