Name: Roll No:

National Institute of Technology Calicut
Department of Computer Science and Engineering
CS2005 Data Structures and Algorithms

Time: 1 Hour **Second Midterm Examination, March 2014** Maximum Marks : 20

20

(Note: For all the questions given below write your answers only in the space provided in the question paper. Answers written elsewhere will not be evaluated)

Part A

1. What would be the running time of the Quicksort algorithm when executed on a sorted array if the pivot element is always chosen as the middle (or one of the middle) element(s) of the subarray being sorted

a) $\theta(n)$ b) $\theta(n^2)$ ✓c) $\theta(n\log n)$ d) None of these. 1Mark

2. Which of the following statements is not correct?

a) Counting sort runs in $\theta(n+k)$ time in the worst case, for n inputs in the range 0 to k.
✓b) Counting sort is a comparison based sorting technique.
c) Counting sort is a stable sorting algorithm.
d) Counting sort can be used in Radix sort for sorting the digits. 1Mark

3. In a singly linked list, the running time of a deletion operation of an element x, a pointer to which is given, is θ(...n.....) and in a doubly linked list the running time of a deletion operation of an element x ,a pointer to which is given, is θ(...1.....). 1Mark

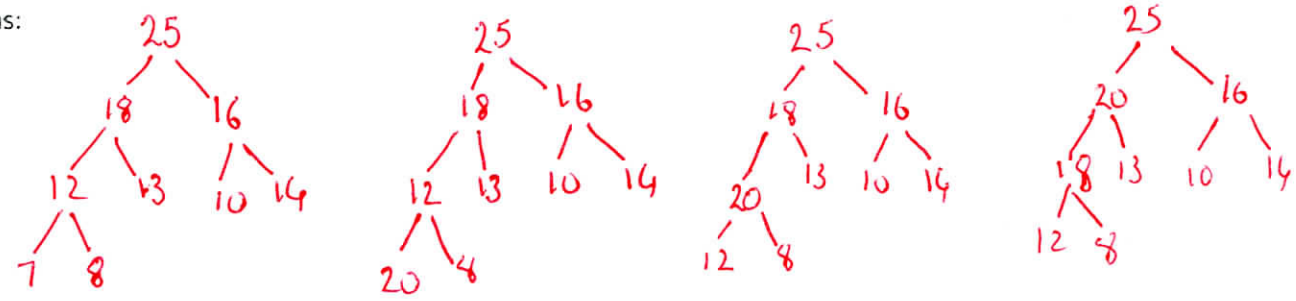
4. In an open addressed hash table, assuming simple uniform hashing, which of the following is the expected running of a successful search operation.(α is the load factor)

a) $\theta(1+\alpha)$ b) $\theta(n)$ c) $\frac{1}{1-\alpha}$ ✓d) $\frac{1}{\alpha}\ln\frac{1}{1-\alpha}$ e) None of these 1Mark

5. The worst case running time of a DELETE operation in a priority queue implemented using heap is θ(..log n..) , and in a priority queue implemented usingdoubly linked list is θ(....1.....). 1Mark

6. Illustrate the operation of HEAP-INCREASE-KEY(A, 8 ,20) applied to the following Max-heap A = < 25, 18,16,12,13,10,14,7,8> by drawing the resulting heap? 1Mark

Ans:



Name:

Roll No:

7. The following keys (24 , 6 , 12 , 3, 4 ,8) are to be inserted into a hash table of size 10 using open addressing and double hashing. The auxiliary hash function is k mod 12 and secondary hash function is $\lfloor m(0.2k \bmod 1) \rfloor$, where m is the size of the hash table. Show the contents of the hash table after the insertion of the keys. 1Mark

a)

0	1	2	3	4	5	6	7	8	9
24		4	3	12		6		8	

b) Show the contents of the hash table after inserting the above set of keys if you are using Quadratic probing with $c_1 = 1$ and $c_2 = 3$. 1Mark

0	1	2	3	4	5	6	7	8	9
24		8	3	12		6		4	

Part B

8. Given two queues A and B is it possible to implement a stack, using only those? If yes, write an algorithm for push(), pop() and is_empty_stack() using the two queues. 3 Marks

```
Push(A,B,x)
1. if (is_queue_empty(A))
2.     enqueue(A,x)
3. else enqueue(B,x)
4. while(!(is_queue_empty(A)))
5.     y = dequeue(A)
6.     enqueue(B,y)
7. // Rename A as B and B as A.
```

```
pop(A,B)
1. dequeue(A)
```

```
is_empty_stack(A,B)
1. return is_queue_empty(A)
```

Name: Roll No:

9. Consider the following pseudo code. (A stack S and a queue Q are used in the code. You may assume that the stack and Queue are initially empty.)

Input : A sequence of n characters. 2 Marks

Output: .....

Unknown()

```
1. for i=1 to n
2.       read (x)
3.       push (S,x)
4.       enqueue(Q,x)
5. While(!Stack_Empty(S))
6.       if(pop(S) != dequeue(Q))
7.                return(nil)
8  return(True)
```

a) What does the algorithm Unknown do?

The algorithm returns True if the entered sequence of characters forms a palindrome otherwise it returns NIL.

b) Give one example sequence for the case for which algorithm Unknown() returns nil and one example for which Unknown() returns True.

ENGLISH → returns nil

MALAYALAM → returns True.

10. Consider a hash table T of size m, if n keys are to be inserted in to the table assuming simple uniform hashing, what the expected number of collisions? Give reasons for your answer.

1Mark

Assuming simple uniform hashing, every key has equal probability of hashing into a slot $= 1/m$

$\therefore$ The expected number of collisions $= \sum_{i=1}^{n} \frac{1}{m} = \frac{n}{m} = \alpha$

Name: Roll No:

Part C

11. Consider the following algorithm A:

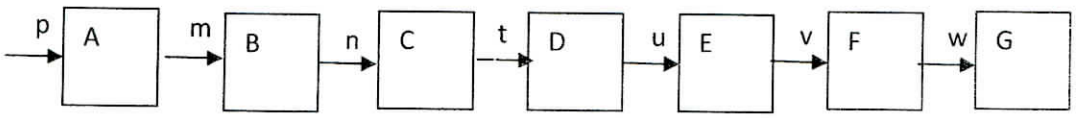
Input: A pointer p to node in a singly linked list

```
A(p)
If ((p==NIL) OR (p.next==NIL))
        return(p)
        else
        q=p.next
        start=q
        while(q!=NIL)
                r=q.next
                q.next=p
                p.next=r
                p=r
                if (p==NIL)
                        q=NIL
                        else
                        q=p.next
        return(start)
```

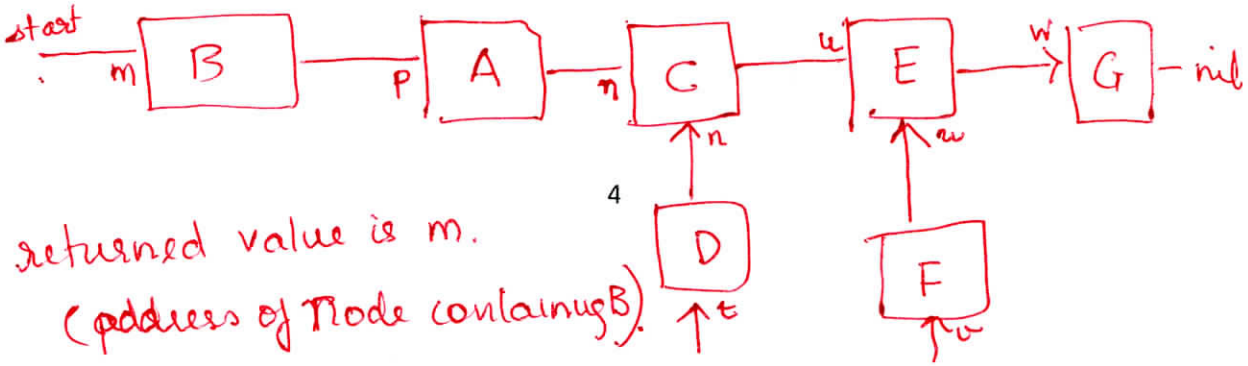
a. State what the algorithm p does in less than 50 words 1.5 Marks

If the list has two or more elements, then the pointer to second node is returned and the list structure is changed as follows: The order of the first two elements are changed, thereafter for every pair of elements, the second is delinked from the list & made to point to the first of the pair.

b. For the input p shown, draw the linked list after A is executed and state the return value.



1.5 Marks



returned value is m.

(address of Node containing B).

Name: Roll No:

12. a. Write an algorithm to delete every third element from a given linked list. For example, if the linked list had been containing the nodes 2, 4, 5, 3, 7, 8, 9, 14, the resulting linked list should contain 2, 4, 3, 7, 9, 14. The pointer to the first node alone is given. Your algorithm should take care of boundary conditions. 2 Marks

```
third-delete(X)
1. if ((X != NIL) AND (X.next != NIL))
2.         p = X
3.         q = X.next
4.         while ((q != NIL) AND (q.next != NIL))
5.                 r = q.next
6                  q.next = r.next
7.                 p = q.next
8.                 if (p == NIL) q = NIL
9.                 else q = p.next
10 return(X)
```

b. State the loop invariant for the loop in your algorithm. 1 Mark

At the beginning of every while iteration, the elements from X (start element) to p are such that the sub list contains elements with all the third elements removed, and all elements from the first non null element in list after p, to the end of the list, are as in the original list