Quiz-1
Data Structures

Set A

23rd Jan Time Allowed: **45** minutes

INSTRUCTIONS

1. This paper contains **Multiple choice questions** .
2. Marking Scheme

   All options marked correctly +5 points

   Correct option not marked / wrong option marked -3 points

   Question not attempted -2 points .
3. All questions carry equal marks.
4. Roll No and answers to be marked on last page .

---

1. Let findMin be a stack operation that reports the minimum element in the current stack. You were to design a stack that permits three operations: push, pop, findMin. What would be the time and space complexity of the findMin operation.

   (a) O(n) Extra Time and O(n) Extra Space

   (b) O(n) Extra Time and O(1) Extra Space

   (c) O(1) Extra Time and O(n) Extra Space

   (d) O(1) Extra Time and O(1) Extra Space

2. Consider any two functions $f(n)$ and $g(n)$, which of the following are true:

   (a) $f(n) + g(n) = \theta(maxf(n), g(n))$
   (b) $f(n) + g(n) = \theta(minf(n), g(n))$
   (c) $f(n) + g(n) = O(maxf(n), g(n))$
   (d) $f(n) + g(n) = O(minf(n), g(n))$

3. which of the following are true:

   (a) $\sum_{i=1}^{n} 1/i = \theta(logn)$
   (b) if $f(n) = \theta(g(n))$ and $g(n) = \theta(h(n))$, then $h(n) = \theta(f(n))$
   (c) if $f(n) = O(g(n))$ and $g(n) = O(h(n))$, then $h(n) = \Omega(f(n))$
   (d) if $f(n) = O(g(n))$ and $g(n) = O(h(n))$, then $h(n) = g(n)$

4. Given following array of integer:

   2, 7, 5, 14, 88, 103, 99, 150

   Which sorting algorithm should be used to sort given data ?

   (a) Merge Sort (b) Quick Sort (c) Insertion Sort (d) Selection Sort

5. Stack A has the entries a,b,c(with a on top). Stack B is empty.An entry popped out of stack A can be printed immediately or pushed to stack B.An entry popped out of the stack B can only be printed. In this arrangement, which of the following permutations of a,b,c are possible?

   (a) b, a, c (b) b, c, a (c) c, a, b (d) c, b, a

6. Consider the following operation along with Enqueue and Dequeue operations on queues, where k is a global parameter.

```
MultiDequeue(Q)
{
   m = k
   while (Q is not empty and m > 0) {
      Dequeue(Q)
      m = m - 1
   }
}
```

   What is the worst case time complexity of a sequence of n MultiDequeue() operations on an initially empty queue?

(a) $\theta(n)$ (b) $\theta(n+K)$ (c) $\theta(n.K)$ (d) $\theta(n^2)$

7. The postfix form of A*B+C/D is?

   (a) *AB/CD+

   (b) AB*CD/+

   (c) A*BC+/D

   (d) ABCD+/*

8. What is the minimum number of stacks of size n required to implement a queue of size n?

   (a) 1 (b) 2 (c) 3 (d) 4

9. Which of the following recurrence relations can be solved using Master Theorem:

   (a) T(n) = T(n/2) + 2^n

   (b) T(n) = n.T(n/2) + n^2

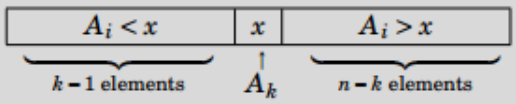
   (c) T(n) = 3.T(n/2) + 4.T(n/6) + $n/2$

   (d) T(n) = 4.T(n/2) + n^2.5

10. Consider the Median of Median Algorithm to find the $i^(th)$ element of an array

Algorithm: SELECT(A, i)

1. Divide the n items into groups of 5 (plus any remainder).
2. Find the median of each group of 5 (by rote). (If the remainder group has an even number of elements, then break ties arbitrarily, for example by choosing the lower median.)
3. Use SELECT recursively to find the median (call it x) of these $\lceil n/5 \rceil$ medians.
4. Partition around x.* Let k = rank(x).†



5. - If $i = k$, then return x.
   - Else, if $i < k$, use SELECT recursively by calling SELECT($A[1,\ldots,k-1]$, i).‡
   - Else, if $i > k$, use SELECT recursively by calling SELECT($A[k+1,\ldots,i]$, $i-k$).

If instead of dividing array in groups of 5 if it was divided into group of 13, which of the following options are correct regarding the recurrence relation and time complexity of the above algorithm .

Hint: try using Substitution method to solve for the recurrence relation .

(a) T(n) = T(n/13)+T(7n/26)+O(n) , O(n)

(b) T(n) = T(n/13)+T(19n/26)+O(n) , O(n)

(c) T(n) = T(n/13)+T(7n/26)+O(n) , O(n^2)

(d) T(n) = T(n/13)+T(19n/26)+O(n) , O(n^2)

11. The following postfix expression with single digit operands is evaluated using a stack:

$$8\ 2\ 3\ \$\ /\ 2\ 3\ *\ +\ 5\ 1\ *\ -$$

Note that $ is the exponent operator. The top two elements of the stack after the first * is evaluated are:

(a) 6,1 (b) 5,7 (c) 3,2 (d) 1,5

12. Assume we have a linear time algorithm that finds the median of an array, which is then used as a pivot in quick sort algorithm, which of the following are true for worst case time complexity of the resultant quick sort algorithm ?

(a) O(N^2) (b) $\theta(N^2)$ (c) O($NlogN$) (d) $\theta(NlogN)$

13. What is the worst case time complexity of insertion sort where position of the data to be inserted is calculated using binary search?

(a) O(N) (b) O(N^2) (c) O($Nlog(N)$) (d) O($N(log(N)^2)$)

14. Solve the recurrence relation
T(n) = 2T(n-1) + T(n-2) + 1

(a) $O(n^3)$ (b) $O(2^n)$ (c)$O(3^n)$ (d) $O(4^n)$

15. Which of the following is not a stable sorting algorithm in its typical implementation. An algorithm is stable if the relative ordering of equal elements doesn't change in the original and the sorted elements.

(a) Insertion sort (b) Merge sort (c) Quick sort (d) Bubble sort

16. Consider the following recurrence. T(n) = T($\sqrt{n}$) + $\Theta(loglogn)$ What is the value of recurrence?

    (a) $\theta((loglogn)^2)$ (b) $\theta(loglogn)$ (c) $\theta(n)$ (d) $\theta(logloglogn)$

17. What is the worst case time complexity of insertion sort where position of the data to be inserted is calculated using binary search?

    (a) O(N) (b) O(N^2) (c) O($Nlog(N)$) (d) O($N(log(N)^2)$)

18. An unordered list contains n distinct elements. The number of comparisons to find an element in this list that is neither maximum nor minimum by the most optimal algorithm is ?

    (a) $\theta(n)$ (b) $\theta(logn)$ (c) $\theta(nlog(n))$ (d) $\theta(1)$

19. An unordered list contains n distinct elements. The number of comparisons to find an element in this list that is neither maximum nor minimum by the most optimal algorithm is ?

    (a) $\theta(n)$ (b) $\theta(logn)$ (c) $\theta(nlog(n))$ (d) $\theta(1)$

20. Which of the following is false ?

    (a) 100nlog(n) = O(nlog(n)/100) (c) $\sqrt{log(n)} = log(log(n))$

    (b) If 0 <x<y then n^x is $O(n^y)$ (d) $2^n \neq O(nk)$

Roll No : _____
Seat No: _____

Answers:

	A	B	C	D
1	①	②	③	④
2	①	②	③	④
3	①	②	③	④
4	①	②	③	④
5	①	②	③	④
6	①	②	③	④
7	①	②	③	④
8	①	②	③	④
9	①	②	③	④
10	①	②	③	④
11	①	②	③	④
12	①	②	③	④
13	①	②	③	④
14	①	②	③	④
15	①	②	③	④
16	①	②	③	④
17	①	②	③	④
18	①	②	③	④