

Time Bound Paper #1 Algorithms

Maximum Time : 70 Minutes

Maximum Marks : 40

1. There are total 25 Questions carrying 40 Marks.
2. Question 1 through 10 are 1-marks question, question 11-25 are 2-marks questions.
3. Question pair (Q. 22 and Q. 23) are linked answer question. The answer to the second question of the above pair depends on the answer to first question of the pair. If the first question in the linked pair is wrongly answered or is un-attempted, then the answer to the second question in the pair will not be evaluated.
4. Wrong answer will carry NEGATIVE marks. For Q 1-10, 1/3 marks for each wrong answer. For Q 11-25, 2/3 marks for each wrong answer.

Prepared by Gatehelpdesk group

www.gatehelpdesk.com

**Q(1-10) carry one marks each**

1.  $o(g(n)) \cap \omega(g(n))$  is:  
A)  $o(g(n))$ .  
B)  $\omega(g(n))$ .  
C) null set.  
D) none of them.
2. Which one is not 'in place' sorting technique:  
A) Merge Sort.  
B) Quick Sort.  
C) Heap Sort.  
D) Insertion Sort.
3. There are  $n$  numbers of students in a class. We want to find the student who got third highest marks in the class. Which sorting technique would be best?  
A) Insertion Sort.  
B) Quick sort.  
C) Bubble Sort.  
D) Heap Sort.
4. Which sorting technique uses incremental approach?  
A) Insertion Sort.  
B) Merge Sort.  
C) Quick Sort.  
D) Radix Sort.
5. If  $f(n) = O(g(n))$  and  $g(n) = O(h(n))$  then :  
A)  $f(n) = \Theta(h(n))$ .  
B)  $f(n) = O(h(n))$ .  
C)  $f(n) = \Omega(h(n))$ .  
D)  $f(n) = o(h(n))$ .
6. The growth of Fibonacci number is:  
A) polynomial.  
B) Exponential.  
C) Linear.  
D) Constant.
7. Which one is not a NP-Complete problem?  
A) Clique Number.  
B) Vertex Cover.  
C) Traveling Salesman Problem.  
D) Matrix Chain Multiplication.

8. 2 persons A & B want to prove that a problem  $\Pi$  is NP Complete. A prove that  $3 \text{ SAT} \leq_p \Pi$  ( $\Pi$  is in base). And B prove that  $\Pi \leq_p 3\text{SAT}$ .

Which one is true?

- A)  $\Pi$  is NP.
- B)  $\Pi$  is NP hard.
- C)  $\Pi$  is NP Complete.
- D) None.

9. An Advantages of chained hash table over the open addressing scheme is:

- A) worst case complexity of search operation is less.
- B) Space used is less.
- C) Deletion is easier.
- D) None of the above.

10. Heap Allocation is required for languages

- A) That supports recursion.
- B) That support Dynamic Data Structures.
- C) That use Dynamic Scope rules.
- D) None of the above.

**Q(11-25) carry two marks each**

11. Consider an array  $A[1..n]$ . It consists of a permutation of numbers  $1..n$ . Now compute another array  $B[1..n]$  as follows:  $B[A[i]] = i$  for all  $i$ . which of the following is true?

- A) B will be a sorted array.
- B) B is a permutation of array A.
- C) Doing the same transformation twice will not give the same array.
- D) B is not a permutation of A.
- E) None of the above.

12. Consider the class of recursive and iterative programs. Which of the following is false?

- A) Recursive program are more powerful then iterative program.
- B) For every iterative program there is an equivalent recursive program.
- C) Recursive program require dynamic memory management.
- D) Recursive program do not terminates sometimes.
- E) Iterative program and recursive program are equally expressive.

13. The first  $n$  cells of an array  $L$  contains positive integers sorted in decreasing order, and the remaining  $m-n$  cells all contain 0. Then, given an integer  $x$ , in how many comparisons can one find the position of  $x$  in  $L$ .

- A) At least  $n$  comparisons are necessary in worst case.
- B) At least  $\log m$  comparisons are necessary in worst case.
- C)  $O(\log(m-n))$  comparisons suffice.
- D)  $O(\log n)$  comparisons suffice.
- E)  $O(\log(m/n))$  comparisons suffice.

14. Suppose you are given  $n$  numbers and you sort them in descending order as follow: First find the maximum. Remove this element from the list and find the maximum of remaining elements. Remove this element and so on, until all the elements exhausted. How many comparisons does this method require in worst case?

- A)  $O(n^2)$ .
- B)  $O(n \log n)$ .
- C) Same as heap sort.
- D)  $O(n)$ .

15. Suppose you are given an array  $A$  with  $2n$  numbers. The numbers in odd positions are sorted in ascending order, that is,  $A[1] \leq A[3] \leq \dots \leq A[2n-1]$ . The number at even positions are sorted in descending order, that is,  $A[2] \geq A[4] \geq \dots \geq A[2n]$ . what method you would recommended for determining if a given number is in the array?

- A) Sort the array using quick sort and the perform binary search.
- B) Merge sorted lists and perform binary search.
- C) Perform a single binary search.
- D) Perform a separate binary searches on the odd positions and even positions.
- E) Perform a sequentially from the end of the array.

16. Consider sorting  $n$  number stored in array  $A$  by first finding the smallest element of  $A$  and exchanging it with the element in  $A[1]$ . Then find the second smallest element of  $A$  and exchange it with  $A[2]$ . Continue in this manner for the first  $n-1$  elements of  $A$ . what will be the time complexity?

- E)  $O(n^2)$ .
- F)  $O(n \log n)$ .
- G)  $O(n^2 \log n)$ .
- H)  $O(n)$ .

17. To solve factorial of a number we have designed an recursive approach given below:

```

Recursion(int x)
{
  If( $x \leq 1$ )
  {
    Return 1;
  }
  Else
  {
    F= $x$ *-----
    Return f;
  }
}

```

One statement in this program is missing. Select best appropriate among the options:

- A) `recursion(x*x)`
- B) `recursion(x)`
- C) `recursion(x-1)`

D) recursion(x+1)

18. Find the complexity of this recursive relation:

$$T(n) = \begin{cases} 2T(n/2) + n & \text{if } n > 1 \\ 1 & \text{if } n = 1 \end{cases}$$

- A)  $O(n \log n)$ .
- B)  $O(n)$ .
- C)  $O(\log n)$ .
- D)  $O(n^2)$ .

19. consider the following two functions:

- i)  $g'(n) = \begin{cases} n^3 & \text{for } 0 \leq n < 10,000 \\ n^2 & \text{for } n \geq 10,000 \end{cases}$
- ii)  $g''(n) = \begin{cases} n & \text{for } 0 \leq n < 100 \\ n^3 & \text{for } n > 100 \end{cases}$

Which one of following is true?

- A)  $g'(n)$  is  $O(g''(n))$ .
- B)  $g'(n)$  is  $O(n^3)$ .
- C)  $g''(n)$  is  $O(g'(n))$ .
- D)  $g''(n)$  is  $O(g'(n))$ .

20. Let  $f(n)$ ,  $g(n)$  and  $h(n)$  be the functions defined for positive integers such that:  $F(n) = O(g(n))$ ,  $g(n) \neq O(f(n))$ ,  $g(n) = O(h(n))$  and  $h(n) = O(g(n))$ . which of the following statement is false?

- A)  $f(n) + g(n) = O(h(n)) + h(n)$
- B)  $h(n) \neq O(f(n))$
- C)  $f(n)h(n) \neq O(g(n))h(n)$
- D) None

21. The time complexity of following C function is (assume  $n > 0$ ):

```
int recursive(int n)
{
    if(n==1)
        return (1);
    else
        return recursive(n-1)+recursive(n-1);
}
```

- A)  $O(n)$
- B)  $O(n \log n)$
- C)  $O(n^2)$
- D)  $O(n^3)$

### **Linked answer Questions**

#### **Statement for Linked Questions Q(22-23)**

Suppose we have  $n/k$  sub lists of length  $k$  are sorted using insertion sort and then merged using the standard merging mechanism

22. What will be the worst case time complexity for merging the lists ?

- A)  $\Theta(nk)$
- B)  $\Theta(n \lg(n/k))$
- C)  $\Theta(\lg(n/k))$
- D)  $\Theta((n/k)^2)$

23. What will be the overall worst case time complexity?

- A)  $\Theta((n/k)^2 + nk)$
- B)  $\Theta(nk \lg(n/k))$
- C)  $\Theta(nk + n \lg(n/k))$
- D) None

24. Which of the following algorithm is not NP-hard?

- A) Hamilton Circuit Problem.
- B) The 0/1 Knapsack Problem.
- C) Finding biconnected components of a Graph.
- D) The Graph Coloring Problem.

25. What is space complexity?

- A) amount of memory which is needed to run the program.
- B) amount of memory which is needed to store the program.
- C) amount of hard disk which is needed to run the program.
- D) amount of hard disk which is needed to store the program.