Design and Analysis of Algorithms
CSE 4th Sem (PCC CS 404)

Module 1

1.What is the time complexity of insert(index) method in ArrayList
(a) O(n) (b) $O(n^2)$ (c) O(nlogn) (d)O(logn)

2.Space complexity of an algorithm is the maximum amount of _____ required by it during execution.
(a)Time (b)operation (c)Memory Space (d) None of these

3.Frequently, the memory space required by an algorithm is a multiple of the size of input. State if the statement is True or False or Maybe.
(a)True (b) False (c) May be (d) None of these

4.To verify whether a function grows faster or slower than the other function, we have some asymptotic or mathematical notations, which is_____.
(a)Big Omega Ω (f) (b) Big Theta θ (f) (c)Big Oh O (f) (d)All of these

5. Which of the given options provides the increasing order of asymptotic complexity of functions f1, f2, f3 and f4? f1(n) = 2^n, f2(n) = n^(3/2) ,f3(n) = nLogn ,f4(n) = n^(Logn) Select one:
(a) f3, f2, f1, f4 (b)f2, f3, f1, f4 (c) f2, f3, f4, f1 (d) f3, f2, f4, f1

6. Which case of Master's theorem is applicable in the recurrence relation T(n)=0.5*T(n/2)+1/n?
(a) Case 3 (b) Case 1 (c)Master's theorem is not applicable Correct (d)Case 2

7. Define the implementation level 0f data structure?
(a) abstract level (b)implementation level (c) application level (d)none of these

8.In computer science, algorithm refers to a special method usable by a computer for the solution to a problem.
(a) True (b) False

9.Which of the following is incorrect?
Algorithms can be represented:
(a) as pseudo codes (b) as syntax (c) as programs (d) as flowcharts

10._____ comparisons required to sort the list 1, 2, 3.....n using insertion sort.
(a) $(n^2 + n + 2) / 2$ (b) $(n^3 + n – 2) / 2$ (c) $(n^2 + n – 2) / 2$ (d) $(n^2 – n – 2) / 2$

11. The complexity of Fibonacci series is _____
(a) $O(2^n)$ b) O(log n) (c) $O(n^2)$ (d) O(n log n)

12.The big-theta notation for $f(n) = n\log(n^2 + 1) + n^2\log n$ is?
(a) $n^2\log n$ (b) n^2 (c) logn (d) $n\log(n^2)$

13.Master's theorem can be applied on which of the following recurrence relation?
(a) T (n) = 2T (n/2) + 2^n (b) T (n) = 2T (n/3) + sin(n)
(c) T (n) = T (n-2) + 2n^2 + 1 (d) None of these

14.T(n) = 3T(n/2+ 47) + 2n^2 + 10*n − 1/2. T(n) will be
(a) O(n^2) (b) O(n^(3/2)) (c) O(n log n) (d) None of these

15.Solve the following recurrence equation T(n) = 2T(n/2) + n.

16. Solve using Substitution Method T(n) = T(√n) + 1

17. Represent in O notation of following recurrence equation T(n) = 2T(n/2) + √n.
18. State Master Theorem with example.
19. . Write a recurrence for the running time of g(n), and solve that recurrence. Assume that addition can be done in constant time.

```
def g(n):
    if n == 1:
        return 1
    else:
        x = g(n-1)
        return x+x
```

20. Now assume addition takes time Θ(b) where b is the number of bits in the larger number. Write a new recurrence for the running time of g(n), and solve that recurrence. Express your final answer in Θ-notation.

Module 2

1. Where is linear searching used?
(a) When the list has only a few elements
(b) When performing a single search in an unordered list
(c) Used all the time
(d) When the list has only a few elements and When performing a single search in an unordered list

2. What is the best case and worst case complexity of ordered linear search?
(a) O(nlogn), O(logn) (b) O(logn), O(nlogn)
(c) O(n), O(1) (d) O(1), O(n)

3. In which of the cases uniform binary search fails compared to binary search?
(a) A table lookup is generally faster than an addition and a shift
(b) Many searches will be performed on the same array
(c) Many searches will be performed on several arrays of the same length
(d) Complexity of code

4. Given, arr = {1,3,5,6,7,9,14,15,17,19} key = 17 and delta = {5,3,1,0}
How many key comparisons are made?(exclude the comparison used to decide the left or right sub array)
(a) 4 (b) 3 (c) 5 (d) 6

5. What will be the auxiliary space complexity of the following code?

#include<bits/stdc++.h>

```
using namespace std;

void func(char* str2, char* str1)
{
        int m = strlen(str2);
        int n = strlen(str1);
        for (int i = 0; i <= n - m; i++)
        {
                int j;

                for (j = 0; j < m; j++)
                        if (str1[i + j] != str2[j])
                                break;

                if (j == m)
                        cout << i << endl;
        }
}

int main()
{
        char str1[] = "1253234";
        char str2[] = "323";
        func(str2, str1);
        return 0;
}
```

(a) O(n) (b) O(1) (c) O(log n) (d) O(m)

6. Which of the following is correct with regard to insertion sort?
(a) insertion sort is stable and it sorts In-place
(b) insertion sort is unstable and it sorts In-place
(c) insertion sort is stable and it does not sort In-place
(d) insertion sort is unstable and it does not sort In-place

7. The worst case time complexity of insertion sort is $O(n^2)$. What will be the worst case time complexity of insertion sort if the correct position for inserting element is calculated using binary search?
(a) O(nlogn) (b) $O(n^2)$ (c) O(n) (d) O(logn)

8. Which of the following sorting algorithm is best suited if the elements are already sorted?
(a) Heap Sort (b) Quick Sort (c) Insertion Sort (d) Merge Sort

9. What is an external sorting algorithm?
(a) Algorithm that uses tape or disk during the sort
(b) Algorithm that uses main memory during the sort
(c) Algorithm that involves swapping
(d) Algorithm that are considered 'in place'

10. What is the worst case complexity of bubble sort?
(a) O(nlogn) (b) O(logn) (c) O(n) (d) $O(n^2)$

11. What is the worst case time complexity of merge sort?
(a) O(n log n) (b) $O(n^2)$ (c) $O(n^2 \log n)$ (d) $O(n \log n^2)$

12. What is the average case complexity of QuickSort?
(a) O(nlogn) (b) O(logn) (c) O(n) (d) $O(n^2)$

13. What is the best case time complexity randomized quick sort?
(a) O(log n) (b) O(n log n) (c) $O(n^2)$ (d) $O(n^2 \log n)$

14. What is the median of three techniques in quick sort?
(a) quick sort with random partitions
(b) quick sort with random choice of pivot
(c) choosing median element as pivot
(d) choosing median of first, last and middle element as pivot

15. How many elements can be sorted in O(logn) time using Heap sort?
(a) O(1) (b) O(n/2) (c) O(logn/log(logn)) (d) O(logn)

16. Analyze the time complexity of quick sort using recurrence equation.
17. Give Strassen's Matrix Multiplication algorithm.
18. Suppose we are sorting an array of eight integers using heapsort, and we have just finished some heapify (either maxheapify or minheapify) operations. The array now looks like this: 16 14 15 10 12 27 28. Compute number of heapify operations have been performed on root of heap?
19. Describe Divide and conquor paradigm with example.
20. Give algorithm and comparision of Linear Search and Binary Search.

Module 3
1.Which of the following algorithms is the best approach for solving Huffman codes?
(a) exhaustive search (b) greedy algorithm
(c) brute force algorithm (d) divide and conquer algorithm

2. What is the running time of the Huffman encoding algorithm?
(a) O(C) (b) O(log C) (c) O(C log C) (d) O(N log C)

3.What is the running time of the Huffman algorithm, if its implementation of the priority queue is done using linked lists?
(a) O(C) (b) O(log C) (c) O(C log C) (d) $O(C^2)$

4.Which of the following statement about 0/1 knapsack and fractional knapsack problem is correct?
(a) In 0/1 knapsack problem items are divisible and in fractional knapsack items are indivisible
(b) Both are the same
(c) 0/1 knapsack is solved using a greedy algorithm and fractional knapsack is solved using dynamic programming
(d) In 0/1 knapsack problem items are indivisible and in fractional knapsack items are divisible

5.Time complexity of fractional knapsack problem is _____
(a) O(n log n) (b) O(n) (c) $O(n^2)$ (d) O(nW)

6.Given items as {value,weight} pairs {{40,20},{30,10},{20,5}}. The capacity of knapsack=20. Find the maximum value output assuming items to be divisible.
(a) 60 (b) 80 (c) 100 (d) 40

7.Given items as {value,weight} pairs {{60,20},{50,25},{20,5}}. The capacity of knapsack=40. Find the maximum value output assuming items to be divisible and nondivisible respectively.
(a) 100, 80 (b) 110, 70 (c) 130, 110 (d) 110, 80

8.Which of the following is/are property/properties of a dynamic programming problem?
(a) Optimal substructure (b) Overlapping subproblems
(c) Greedy approach (d) Both optimal substructure and overlapping subproblems

9.Which of the following problems is NOT solved using dynamic programming?
(a) 0/1 knapsack problem (b) Matrix chain multiplication problem
(c) Edit distance problem (d) Fractional knapsack problem

10.You are given a knapsack that can carry a maximum weight of 60. There are 4 items with weights {20, 30, 40, 70} and values {70, 80, 90, 200}. What is the maximum value of the items you can carry using the knapsack?
(a) 160 (b) 200 (c) 170 (d) 90

11.Consider the matrices P, Q and R which are 10 x 20, 20 x 30 and 30 x 40 matrices respectively. What is the minimum number of multiplications required to multiply the three matrices?
(a) 18000 (b) 12000 (c) 24000 (d) 32000

12.Which of the following is the recurrence relation for the matrix chain multiplication problem where mat[i-1] * mat[i] gives the dimension of the ith matrix?
(a) dp[i,j] = 1 if i=j
dp[i,j] = min{dp[i,k] + dp[k+1,j]}
(b) dp[i,j] = 0 if i=j
dp[i,j] = min{dp[i,k] + dp[k+1,j]}
(c) dp[i,j] = 1 if i=j
dp[i,j] = min{dp[i,k] + dp[k+1,j]} + mat[i-1]*mat[k]*mat[j].
(d) dp[i,j] = 0 if i=j
dp[i,j] = min{dp[i,k] + dp[k+1,j]} + mat[i-1]*mat[k]*mat[j].

13.Find the length of the longest increasing subsequence for the given sequence:
{-10, 24, -9, 35, -21, 55, -41, 76, 84}
(a) 5 (b) 4 (c) 3 (d) 6

14.Given a rod of length n and the selling prices of all pieces smaller than equal to n, find the most beneficial way of cutting the rod into smaller pieces. This problem is called the rod cutting problem. Which of these methods can be used to solve the rod cutting problem?
(a) Brute force
(b) Dynamic programming
(c) Recursion
(d) Brute force, Dynamic programming and Recursion

15.Which of the following is an application of the edit distance problem?
(a) Approximate string matching
(b) Spelling correction
(c) Similarity of DNA
(d) Approximate string matching, Spelling Correction and Similarity of DNA

16. Differentiate among Divide and conquor, Greedy and Dynamic Programming paradigm.
17. Describe Travelling Salesman Problem.
18. Give the algorithm to compute Fibonacci series using Dynamic programming approach and compare with other approach.
19. Describe Bin packing problem with the help of numerical example.
20. Describe characteristics of Dynamic Programming.

Module 4

1. Depth First Search is equivalent to which of the traversal in the Binary Trees?
(a) Pre-order Traversal (b) Post-order Traversal
(c) Level-order Traversal (d) In-order Traversal

2.Time Complexity of DFS is? (V – number of vertices, E – number of edges)
(a) O(V + E) (b) O(V) (c) O(E) (d) O(V*E)

3. Which of the following is not an application of Depth First Search?
(a) For generating topological sort of a graph
(b) For generating Strongly Connected Components of a directed graph
(c) Detecting cycles in the graph
(d) Peer to Peer Networks

4.Breadth First Search is equivalent to which of the traversal in the Binary Trees?
(a) Pre-order Traversal (b) Post-order Traversal
(c) Level-order Traversal (d) In-order Traversal

5.Time Complexity of Breadth First Search is? (V – number of vertices, E – number of edges)
(a) O(V + E) (b) O(V) (c) O(E) (d) O(V*E)

6.Which of the following is false in the case of a spanning tree of a graph G?
(a) It is tree that spans G (b) It is a subgraph of the G
(c) It includes every vertex of the G (d) It can be either cyclic or acyclic

7.Which of the following is false?
(a) The spanning trees do not have any cycles
(b) MST have n – 1 edges if the graph has n edges
(c) Edge e belonging to a cut of the graph if has the weight smaller than any other edge in the same cut, then the edge e is present in all the MSTs of the graph
(d) Removing one edge from the spanning tree will not make the graph disconnected

8.What is the time complexity of Kruskal's algorithm?
(a) O(log V) (b) O(E log V) (c) $O(E^2)$ (d) O(V log E)

9.Consider the following statements.
S1. Kruskal's algorithm might produce a non-minimal spanning tree.
S2. Kruskal's algorithm can efficiently implemented using the disjoint-set data structure.
(a) S1 is true but S2 is false (b) Both S1 and S2 are false
(c) Both S1 and S2 are true (d) S2 is true but S1 is false

10. Worst case is the worst case time complexity of Prim's algorithm if adjacency matrix is used?
(a) O(log V) (b) $O(V^2)$ (c) $O(E^2)$ (d) O(V log E)

11.Which of the following is false about Prim's algorithm?
(a) It is a greedy algorithm
(b) It constructs MST by selecting edges in increasing order of their weights
(c) It never accepts cycles in the MST
(d) It can be implemented using the Fibonacci heap

12.Dijkstra's Algorithm cannot be applied on _____
(a) Directed and weighted graphs (b) Graphs having negative weight function
(c) Unweighted graphs (d) Undirected and unweighted graphs

13. What is running time of Dijkstra's algorithm using Binary min- heap method?
(a) O(V) (b) O(VlogV) (c) O(E) (d) O(ElogV)

14.Dijkstra's Algorithm is the prime example for _____
(a) Greedy algorithm (b) Branch and bound
(c) Back tracking (d) Dynamic programming

15.Bellmann Ford Algorithm can be applied for _____
(a) Undirected and weighted graphs (b) Undirected and unweighted graphs
(c) Directed and weighted graphs (d) All directed graphs

16. Floyd Warshall Algorithm can be used for finding _____
(a) Single source shortest path (b) Topological sort
(c) Minimum spanning tree (d) Transitive closure

17. With the help of numerical example differentiate between Dijkstra's algorithm and Bellman Ford algorithm.
18.Give the algorithm to find MST using Kurskal's algorithm.
19. Using strong argumnet show that the a garph having less then n-1 edges is disconnected.
20. What is strongly connected component and give its algorithm to find it?

Module 5

1.The worst-case efficiency of solving a problem in polynomial time is?
(a) O(p(n)) (b) O(p(n log n)) (c) $O(p(n^2))$ (d) O(p(m log n))

2.Problems that can be solved in polynomial time are known as?
(a) intractable (b) tractable (c) decision (d) complete

3._____ is the class of decision problems that can be solved by non-deterministic polynomial algorithms?
(a) NP (b) P (c) Hard (d) Complete

4.Problems that cannot be solved by any algorithm are called?
(a) tractable problems (b) intractable problems
(c) undecidable problems (d) decidable problems

5. To which class does the Euler's circuit problem belong?
(a) P class (b) NP class (c) Partition class (d) Complete class

6. Halting problem is an example for?
(a) decidable problem (b) undecidable problem
(c) complete problem (d) trackable problem

7.How many conditions have to be met if an NP- complete problem is polynomially reducible?
(a) 1 (b) 2 (c) 3 (d) 4

8.To which of the following class does a CNF-satisfiability problem belong?
(a) NP class (b) P class (c) NP complete (d) NP hard

9.Which of the following problems is not NP complete?
(a) Hamiltonian circuit (b) Bin packing
(c) Partition problem (d) Halting problem

10.The choice of polynomial class has led to the development of an extensive theory called _____
(a) computational complexity (b) time complexity
(c) problem complexity (d) decision complexity

11.Hamiltonian path problem is _____
(a) NP problem (b) N class problem
(c) P class problem (d) NP complete problem

12. Subset sum problem is an example of NP-complete problem.
(a) true (b) false

13. What is a subset sum problem?
(a) finding a subset of a set that has sum of elements equal to a given number
(b) checking for the presence of a subset that has sum of elements equal to a given number and printing true or false based on the result
(c) finding the sum of elements present in a set
(d) finding the sum of all the subsets of a set

14.What is the set partition problem?
(a) finding a subset of a set that has sum of elements equal to a given number
(b) checking for the presence of a subset that has sum of elements equal to a given number
(c) checking whether the set can be divided into two subsets of with equal sum of elements and printing true or false based on the result
(d) finding subsets with equal sum of elements

15.Set partition problem is an example of NP complete problem.
(a) true (b) false

16. Describe optimization and decision problem.
17. Describe P and NP – class problem.
18. State Cook's Theorem.
19. Explain the different ways to analyze randamize algorithm.
20. Explain the need of approximation algorithm.

Darbhanga College of Engineering Darbhanga

Midterm Exam, March 2020

Design & Analysis of Algorithm (CSE 4th Sem)

Duration: 2 Hrs **Max Marks : 20**

Regestration No:

Instruction:

1. **If any rough work found on question paper during examination, invigilator can deduct 5 marks.**

2. Attempt all questions. No any clarification related to question will be entertained, If you are considering any assumption must mention it.

S. No	Questions	Marks	CO[1]	BL[2]
1.	Solve the recurrence $\mathrm{T}(n) = 2T(n/4) + \sqrt{n}$ OR Apply Master theorm on following recurrence. a. $\mathrm{T}(n) = 9T(n/3) + n$ b. $\mathrm{T}(n) = 3T(n/4) + n\log n$	4	CO1	L1
2.	Show that. a. $\log(!n) = \Theta(n\log n)$. b. For any real coonstants a and b, $b \geq 1$ and $(n+a)^b = \Theta(n^b)$ OR a. Solve the recurrence $\mathrm{T}(n) = T(n-1) + 1/n$	4	CO1	L2
3.	Prove that, For any two function $f(n)$ and $g(n)$, we have $f(n) = \Theta(g(n))$ iff $f(n) = O(g(n))$ and $f(n) = \Omega(g(n))$.	4	CO2	L5
4.	For a given sequence analyze with reason which one will be time and space efficient in the case of Insertion Sort, Quick Sort and Merge Sort. Sequence a : 1,2,3,4,5.....,10. Sequence b : 10,9,8,.....,1.	4	CO3	L3
5.	Compute time complexity of Quick Sort in the best , average and worst case.	4	CO3	L5

End

[1]Course Outcome

[2]Bloom Level

DARBHANGA COLLEGE OF ENGINEERING, DARBHANGA
Mid-Sem Examination

Design and Analysis of Algorithms

(CSE 5th Sem 051506)

Max Marks : 20
Max Time : 2 Hrs

Instruction:

Question number 1 to 8 carry equal marks and each having 5 marks, attempt any four from question number 1 to 8.

OR

Attempt only question number 9, it carry 20 marks.

1. Prove that either graphically or analytically $\Theta(g(n)) = O(g(n)) \cap \Omega(g(n))$.

2. Express following in terms of Θ notation, assume 'n' terms.
   a) $1 + 8 + 27 + 64 + \ldots\ldots$
   b) log (!n), here !n represents factorial of n.

3. Compute the soluition of the following recurrence.
   a) $T(n) = 5xT(n/5) + n / \log n$.
   b) $T(n) = 13246\ xT(n/68) + n^2$.

4. For a given sequence analyze with reason which one will be time and space efficient in the case of Insertion Sort, Quick Sort and Merge Sort.
   Sequence a : 1,2,3,4,5.....,10.
   Sequence b : 10,9,8,.....,1.

5.
a. In the implementation of a priority queue, following operation takes what amount of time in O representation.
i. Insert ii Extract Minimum iii. Decrease Key.
b. What will be the run time of Dijkstra algorithm using priority queue implementation.

6. Show that the second smallest of 'n ' elements can be found with n + lg n – 2 comparisions in the worst case.

7. As shown in figure 1 for a given adjacency list representation of a graph, run topological sort on the graph. Answer should consist of a list of the vertices in a particular order

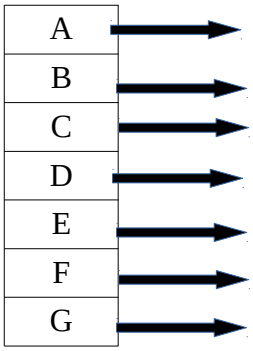

Figure 1

8.
a. Define the following terms with respect to time stamp.
i Tree Edge ii Forward Edge. iii Back Edge iv Cross Edge.
b. Show that a Graph is DAG iff it has no back edge.

9. You are interested in analyzing some hard to obtain data from two separate databases. Each database contains ‘n’ numerical values – so there are 2n values total- and you may assume that no two values are the same. You’d like to determine the median of this set of 2n values, which we will define here to be the n^{th} smallest value.

However, the only way you can access these values is through queries to the databases. In a single query, you can specify a value ‘k’ to one of the two databases, and the chosen database will return the k^{th} smallest value that it contains. Since queries are expensive, you would like to compute the median using as few queries as possible.

Give an algorithm that finds the median value using at most O(log n) queries.

Design and Analysis of Algorithms
CSE 4th Sem (PCC CS 404)

Module 1

1.What is the time complexity of insert(index) method in ArrayList
(a) O(n) (b) $O(n^2)$ (c) O(nlogn) (d)O(logn)

2.Space complexity of an algorithm is the maximum amount of _____ required by it during execution.
(a)Time (b)operation (c)Memory Space (d) None of these

3.Frequently, the memory space required by an algorithm is a multiple of the size of input. State if the statement is True or False or Maybe.
(a)True (b) False (c) May be (d) None of these

4.To verify whether a function grows faster or slower than the other function, we have some asymptotic or mathematical notations, which is_____.
(a)Big Omega Ω (f) (b) Big Theta θ (f) (c)Big Oh O (f) (d)All of these

5. Which of the given options provides the increasing order of asymptotic complexity of functions f1, f2, f3 and f4? f1(n) = 2^n, f2(n) = n^(3/2) ,f3(n) = nLogn ,f4(n) = n^(Logn) Select one:
(a) f3, f2, f1, f4 (b)f2, f3, f1, f4 (c) f2, f3, f4, f1 (d) f3, f2, f4, f1

6. Which case of Master's theorem is applicable in the recurrence relation T(n)=0.5*T(n/2)+1/n?
(a) Case 3 (b) Case 1 (c)Master's theorem is not applicable Correct (d)Case 2

7. Define the implementation level 0f data structure?
(a) abstract level (b)implementation level (c) application level (d)none of these

8.In computer science, algorithm refers to a special method usable by a computer for the solution to a problem.
(a) True (b) False

9.Which of the following is incorrect?
Algorithms can be represented:
(a) as pseudo codes (b) as syntax (c) as programs (d) as flowcharts

10._____ comparisons required to sort the list 1, 2, 3.....n using insertion sort.
(a) $(n^2 + n + 2) / 2$ (b) $(n^3 + n – 2) / 2$ (c) $(n^2 + n – 2) / 2$ (d) $(n^2 – n – 2) / 2$

11. The complexity of Fibonacci series is _____
(a) $O(2^n)$ b) O(log n) (c) $O(n^2)$ (d) O(n log n)

12.The big-theta notation for $f(n) = n\log(n^2 + 1) + n^2\log n$ is?
(a) $n^2\log n$ (b) n^2 (c) logn (d) $n\log(n^2)$

13.Master's theorem can be applied on which of the following recurrence relation?
(a) T (n) = 2T (n/2) + 2^n (b) T (n) = 2T (n/3) + sin(n)
(c) T (n) = T (n-2) + 2n^2 + 1 (d) None of these

14.T(n) = 3T(n/2+ 47) + 2n^2 + 10*n − 1/2. T(n) will be
(a) O(n^2) (b) O(n^(3/2)) (c) O(n log n) (d) None of these

15.Solve the following recurrence equation T(n) = 2T(n/2) + n.

16. Solve using Substitution Method T(n) = T(√n) + 1

17. Represent in O notation of following recurrence equation T(n) = 2T(n/2) + √n.
18. State Master Theorem with example.
19. . Write a recurrence for the running time of g(n), and solve that recurrence. Assume that addition can be done in constant time.

```
def g(n):
    if n == 1:
        return 1
    else:
        x = g(n-1)
        return x+x
```

20. Now assume addition takes time Θ(b) where b is the number of bits in the larger number. Write a new recurrence for the running time of g(n), and solve that recurrence. Express your final answer in Θ-notation.

Module 2

1. Where is linear searching used?
(a) When the list has only a few elements
(b) When performing a single search in an unordered list
(c) Used all the time
(d) When the list has only a few elements and When performing a single search in an unordered list

2. What is the best case and worst case complexity of ordered linear search?
(a) O(nlogn), O(logn) (b) O(logn), O(nlogn)
(c) O(n), O(1) (d) O(1), O(n)

3. In which of the cases uniform binary search fails compared to binary search?
(a) A table lookup is generally faster than an addition and a shift
(b) Many searches will be performed on the same array
(c) Many searches will be performed on several arrays of the same length
(d) Complexity of code

4. Given, arr = {1,3,5,6,7,9,14,15,17,19} key = 17 and delta = {5,3,1,0}
How many key comparisons are made?(exclude the comparison used to decide the left or right sub array)
(a) 4 (b) 3 (c) 5 (d) 6

5. What will be the auxiliary space complexity of the following code?

#include<bits/stdc++.h>

```
using namespace std;

void func(char* str2, char* str1)
{
        int m = strlen(str2);
        int n = strlen(str1);
        for (int i = 0; i <= n - m; i++)
        {
                int j;

                for (j = 0; j < m; j++)
                        if (str1[i + j] != str2[j])
                                break;

                if (j == m)
                        cout << i << endl;
        }
}

int main()
{
        char str1[] = "1253234";
        char str2[] = "323";
        func(str2, str1);
        return 0;
}
```

(a) O(n) (b) O(1) (c) O(log n) (d) O(m)

6. Which of the following is correct with regard to insertion sort?
(a) insertion sort is stable and it sorts In-place
(b) insertion sort is unstable and it sorts In-place
(c) insertion sort is stable and it does not sort In-place
(d) insertion sort is unstable and it does not sort In-place

7. The worst case time complexity of insertion sort is $O(n^2)$. What will be the worst case time complexity of insertion sort if the correct position for inserting element is calculated using binary search?
(a) O(nlogn) (b) $O(n^2)$ (c) O(n) (d) O(logn)

8. Which of the following sorting algorithm is best suited if the elements are already sorted?
(a) Heap Sort (b) Quick Sort (c) Insertion Sort (d) Merge Sort

9. What is an external sorting algorithm?
(a) Algorithm that uses tape or disk during the sort
(b) Algorithm that uses main memory during the sort
(c) Algorithm that involves swapping
(d) Algorithm that are considered 'in place'

10. What is the worst case complexity of bubble sort?
(a) O(nlogn) (b) O(logn) (c) O(n) (d) $O(n^2)$

11. What is the worst case time complexity of merge sort?
(a) O(n log n) (b) $O(n^2)$ (c) $O(n^2 \log n)$ (d) $O(n \log n^2)$

12. What is the average case complexity of QuickSort?
(a) O(nlogn) (b) O(logn) (c) O(n) (d) $O(n^2)$

13. What is the best case time complexity randomized quick sort?
(a) O(log n) (b) O(n log n) (c) $O(n^2)$ (d) $O(n^2 \log n)$

14. What is the median of three techniques in quick sort?
(a) quick sort with random partitions
(b) quick sort with random choice of pivot
(c) choosing median element as pivot
(d) choosing median of first, last and middle element as pivot

15. How many elements can be sorted in O(logn) time using Heap sort?
(a) O(1) (b) O(n/2) (c) O(logn/log(logn)) (d) O(logn)

16. Analyze the time complexity of quick sort using recurrence equation.
17. Give Strassen's Matrix Multiplication algorithm.
18. Suppose we are sorting an array of eight integers using heapsort, and we have just finished some heapify (either maxheapify or minheapify) operations. The array now looks like this: 16 14 15 10 12 27 28. Compute number of heapify operations have been performed on root of heap?
19. Describe Divide and conquor paradigm with example.
20. Give algorithm and comparision of Linear Search and Binary Search.

Module 3
1.Which of the following algorithms is the best approach for solving Huffman codes?
(a) exhaustive search (b) greedy algorithm
(c) brute force algorithm (d) divide and conquer algorithm

2. What is the running time of the Huffman encoding algorithm?
(a) O(C) (b) O(log C) (c) O(C log C) (d) O(N log C)

3.What is the running time of the Huffman algorithm, if its implementation of the priority queue is done using linked lists?
(a) O(C) (b) O(log C) (c) O(C log C) (d) $O(C^2)$

4.Which of the following statement about 0/1 knapsack and fractional knapsack problem is correct?
(a) In 0/1 knapsack problem items are divisible and in fractional knapsack items are indivisible
(b) Both are the same
(c) 0/1 knapsack is solved using a greedy algorithm and fractional knapsack is solved using dynamic programming
(d) In 0/1 knapsack problem items are indivisible and in fractional knapsack items are divisible

5.Time complexity of fractional knapsack problem is _____
(a) O(n log n) (b) O(n) (c) $O(n^2)$ (d) O(nW)

6.Given items as {value,weight} pairs {{40,20},{30,10},{20,5}}. The capacity of knapsack=20. Find the maximum value output assuming items to be divisible.
(a) 60 (b) 80 (c) 100 (d) 40

7.Given items as {value,weight} pairs {{60,20},{50,25},{20,5}}. The capacity of knapsack=40. Find the maximum value output assuming items to be divisible and nondivisible respectively.
(a) 100, 80 (b) 110, 70 (c) 130, 110 (d) 110, 80

8.Which of the following is/are property/properties of a dynamic programming problem?
(a) Optimal substructure (b) Overlapping subproblems
(c) Greedy approach (d) Both optimal substructure and overlapping subproblems

9.Which of the following problems is NOT solved using dynamic programming?
(a) 0/1 knapsack problem (b) Matrix chain multiplication problem
(c) Edit distance problem (d) Fractional knapsack problem

10.You are given a knapsack that can carry a maximum weight of 60. There are 4 items with weights {20, 30, 40, 70} and values {70, 80, 90, 200}. What is the maximum value of the items you can carry using the knapsack?
(a) 160 (b) 200 (c) 170 (d) 90

11.Consider the matrices P, Q and R which are 10 x 20, 20 x 30 and 30 x 40 matrices respectively. What is the minimum number of multiplications required to multiply the three matrices?
(a) 18000 (b) 12000 (c) 24000 (d) 32000

12.Which of the following is the recurrence relation for the matrix chain multiplication problem where mat[i-1] * mat[i] gives the dimension of the ith matrix?
(a) dp[i,j] = 1 if i=j
dp[i,j] = min{dp[i,k] + dp[k+1,j]}
(b) dp[i,j] = 0 if i=j
dp[i,j] = min{dp[i,k] + dp[k+1,j]}
(c) dp[i,j] = 1 if i=j
dp[i,j] = min{dp[i,k] + dp[k+1,j]} + mat[i-1]*mat[k]*mat[j].
(d) dp[i,j] = 0 if i=j
dp[i,j] = min{dp[i,k] + dp[k+1,j]} + mat[i-1]*mat[k]*mat[j].

13.Find the length of the longest increasing subsequence for the given sequence:
{-10, 24, -9, 35, -21, 55, -41, 76, 84}
(a) 5 (b) 4 (c) 3 (d) 6

14.Given a rod of length n and the selling prices of all pieces smaller than equal to n, find the most beneficial way of cutting the rod into smaller pieces. This problem is called the rod cutting problem. Which of these methods can be used to solve the rod cutting problem?
(a) Brute force
(b) Dynamic programming
(c) Recursion
(d) Brute force, Dynamic programming and Recursion

15.Which of the following is an application of the edit distance problem?
(a) Approximate string matching
(b) Spelling correction
(c) Similarity of DNA
(d) Approximate string matching, Spelling Correction and Similarity of DNA

16. Differentiate among Divide and conquor, Greedy and Dynamic Programming paradigm.
17. Describe Travelling Salesman Problem.
18. Give the algorithm to compute Fibonacci series using Dynamic programming approach and compare with other approach.
19. Describe Bin packing problem with the help of numerical example.
20. Describe characteristics of Dynamic Programming.

Module 4

1. Depth First Search is equivalent to which of the traversal in the Binary Trees?
(a) Pre-order Traversal (b) Post-order Traversal
(c) Level-order Traversal (d) In-order Traversal

2.Time Complexity of DFS is? (V – number of vertices, E – number of edges)
(a) O(V + E) (b) O(V) (c) O(E) (d) O(V*E)

3. Which of the following is not an application of Depth First Search?
(a) For generating topological sort of a graph
(b) For generating Strongly Connected Components of a directed graph
(c) Detecting cycles in the graph
(d) Peer to Peer Networks

4.Breadth First Search is equivalent to which of the traversal in the Binary Trees?
(a) Pre-order Traversal (b) Post-order Traversal
(c) Level-order Traversal (d) In-order Traversal

5.Time Complexity of Breadth First Search is? (V – number of vertices, E – number of edges)
(a) O(V + E) (b) O(V) (c) O(E) (d) O(V*E)

6.Which of the following is false in the case of a spanning tree of a graph G?
(a) It is tree that spans G (b) It is a subgraph of the G
(c) It includes every vertex of the G (d) It can be either cyclic or acyclic

7.Which of the following is false?
(a) The spanning trees do not have any cycles
(b) MST have n – 1 edges if the graph has n edges
(c) Edge e belonging to a cut of the graph if has the weight smaller than any other edge in the same cut, then the edge e is present in all the MSTs of the graph
(d) Removing one edge from the spanning tree will not make the graph disconnected

8.What is the time complexity of Kruskal's algorithm?
(a) O(log V) (b) O(E log V) (c) $O(E^2)$ (d) O(V log E)

9.Consider the following statements.
S1. Kruskal's algorithm might produce a non-minimal spanning tree.
S2. Kruskal's algorithm can efficiently implemented using the disjoint-set data structure.
(a) S1 is true but S2 is false (b) Both S1 and S2 are false
(c) Both S1 and S2 are true (d) S2 is true but S1 is false

10. Worst case is the worst case time complexity of Prim's algorithm if adjacency matrix is used?
(a) O(log V) (b) $O(V^2)$ (c) $O(E^2)$ (d) O(V log E)

11.Which of the following is false about Prim's algorithm?
(a) It is a greedy algorithm
(b) It constructs MST by selecting edges in increasing order of their weights
(c) It never accepts cycles in the MST
(d) It can be implemented using the Fibonacci heap

12.Dijkstra's Algorithm cannot be applied on _____
(a) Directed and weighted graphs (b) Graphs having negative weight function
(c) Unweighted graphs (d) Undirected and unweighted graphs

13. What is running time of Dijkstra's algorithm using Binary min- heap method?
(a) O(V) (b) O(VlogV) (c) O(E) (d) O(ElogV)

14.Dijkstra's Algorithm is the prime example for _____
(a) Greedy algorithm (b) Branch and bound
(c) Back tracking (d) Dynamic programming

15.Bellmann Ford Algorithm can be applied for _____
(a) Undirected and weighted graphs (b) Undirected and unweighted graphs
(c) Directed and weighted graphs (d) All directed graphs

16. Floyd Warshall Algorithm can be used for finding _____
(a) Single source shortest path (b) Topological sort
(c) Minimum spanning tree (d) Transitive closure

17. With the help of numerical example differentiate between Dijkstra's algorithm and Bellman Ford algorithm.
18.Give the algorithm to find MST using Kurskal's algorithm.
19. Using strong argumnet show that the a garph having less then n-1 edges is disconnected.
20. What is strongly connected component and give its algorithm to find it?

Module 5

1.The worst-case efficiency of solving a problem in polynomial time is?
(a) O(p(n)) (b) O(p(n log n)) (c) $O(p(n^2))$ (d) O(p(m log n))

2.Problems that can be solved in polynomial time are known as?
(a) intractable (b) tractable (c) decision (d) complete

3._____ is the class of decision problems that can be solved by non-deterministic polynomial algorithms?
(a) NP (b) P (c) Hard (d) Complete

4.Problems that cannot be solved by any algorithm are called?
(a) tractable problems (b) intractable problems
(c) undecidable problems (d) decidable problems

5. To which class does the Euler's circuit problem belong?
(a) P class (b) NP class (c) Partition class (d) Complete class

6. Halting problem is an example for?
(a) decidable problem (b) undecidable problem
(c) complete problem (d) trackable problem

7.How many conditions have to be met if an NP- complete problem is polynomially reducible?
(a) 1 (b) 2 (c) 3 (d) 4

8.To which of the following class does a CNF-satisfiability problem belong?
(a) NP class (b) P class (c) NP complete (d) NP hard

9.Which of the following problems is not NP complete?
(a) Hamiltonian circuit (b) Bin packing
(c) Partition problem (d) Halting problem

10.The choice of polynomial class has led to the development of an extensive theory called _____
(a) computational complexity (b) time complexity
(c) problem complexity (d) decision complexity

11.Hamiltonian path problem is _____
(a) NP problem (b) N class problem
(c) P class problem (d) NP complete problem

12. Subset sum problem is an example of NP-complete problem.
(a) true (b) false

13. What is a subset sum problem?
(a) finding a subset of a set that has sum of elements equal to a given number
(b) checking for the presence of a subset that has sum of elements equal to a given number and printing true or false based on the result
(c) finding the sum of elements present in a set
(d) finding the sum of all the subsets of a set

14.What is the set partition problem?
(a) finding a subset of a set that has sum of elements equal to a given number
(b) checking for the presence of a subset that has sum of elements equal to a given number
(c) checking whether the set can be divided into two subsets of with equal sum of elements and printing true or false based on the result
(d) finding subsets with equal sum of elements

15.Set partition problem is an example of NP complete problem.
(a) true (b) false

16. Describe optimization and decision problem.
17. Describe P and NP – class problem.
18. State Cook's Theorem.
19. Explain the different ways to analyze randamize algorithm.
20. Explain the need of approximation algorithm.