Algorithms Volume-1 Questions

Practice Questions

01. Which of the given options provides the increasing order of asymptotic Complexity of functions f_1, f_2, f_3 and f_4?
$f_1(n) = 2^n$ $f_2(n) = n^{3/2}$ $f_3(n) = n\log_2 n$
$f_4(n) = n^{\log_2 n}$
(a) f_3, f_2, f_4, f_1
(b) f_3, f_2, f_1, f_4
(c) f_2, f_3, f_1, f_4
(d) f_2, f_3, f_4, f_1

02. Which of the following is **false**?
(a) $100\, n\log n = O\left(\frac{n\log n}{100}\right)$
(b) $\sqrt{\log n} = O(\log\log n)$
(c) If $0 < x < y$ then $n^x = O(n^y)$
(d) $2^n \neq O(n^k)$

03. Let $f(n) = n^2 \log n$ and
$g(n) = n(\log n)^{10}$
be two positive functions of n. Which of the following statements is **correct**?
(a) $f(n) = O(g(n))$ and $g(n) \neq O(f(n))$
(b) $g(n) = O(f(n))$ and $f(n) \neq O(g(n))$
(c) $f(n) \neq O(g(n))$ and $g(n) \neq O(f(n))$
(d) $f(n)=O(g(n))$ and $g(n) =O(f(n))$

04. Consider the following three claims
I. $(n + k)^m = \Theta(n^m)$ where k and m are constants
II. $2^{n+1} = O(2^n)$
III. $2^{2n} = O(2^n)$

Which of these claims are **correct**?
(a) I and II
(b) I and III
(c) II and III
(d) I, II, and III

05. Two alternative packages A and B are available for processing a database having 10^k records. Package A requires $0.0001n^2$ time units and package B requires $10n.\log_{10} n$ time units to process n records. What is the smallest value of k for which package B will be preferred over A?
(a) 12 (b) 10
(c) 6 (d) 5

06. Consider the following two functions:
$$g_1(n) = \begin{cases} n^3 \text{ for } 0 \le n < 10{,}000 \\ n^2 \text{ for } n \ge 10000 \end{cases}$$
$$g_2(n) = \begin{cases} n \text{ for } 0 \le n \le 100 \\ n^3 \text{ for } n > 100 \end{cases}$$
Which of the following is **true**?
(a) $g_1(n)$ is $O(g_2(n))$
(b) $g_1(n)$ is $O(n^3)$
(c) $g_2(n)$ is $O(g_1(n))$
(d) g2 (n) is o(g2(n))

07. Consider the following functions
$f(n) = 3n^{\sqrt{n}}$
$g(n) = 2^{\sqrt{n}\log_2 n}$
$h(n) = n!$

Which of the following is **true**?
(a) h(n) is O(f(n)) (b) h(n) is O(g(n))
(c) g(n) is not O(f(n)) (d) f(n) is O(g(n))

08. Consider the following functions:

$f(n) = 2^n$

$g(n) = n!$

$h(n) = n^{\log n}$

Which of the following statements about the asymptotic behaviour of f(n), g(n), and h(n) is **true**?

(a) $f(n) = O(g(n)); g(n) = O(h(n))$

(b) $f(n) = \Omega(g(n)); f(n) = O(h(n))$

(c) $g(n) = O(f(n)); h(n) = O(f(n))$

(d) $h(n) = O(f(n)); g(n) = \Omega(f(n))$

09. Consider the following C code segment:

```
int IsPrime(n)
{
   int i, n;
   for(i=2;i<=sqrt (n);i++)
   if(n%i = = 0)
   {
          printf("Not Prime\n"); return 0;
   }
   return 1;
}
```

Let T(n) denote the number of times the for loop is executed by the program on input n. Which of the following is **TRUE**?

(a) $T(n) = O(\sqrt{n})$ and $T(n) = \Omega(\sqrt{n})$

(b) $T(n) = O(\sqrt{n})$ and $T(n) = \Omega(1)$

(c) $T(n) = O(n)$ and $T(n) = \Omega(\sqrt{n})$

(d) None of the above

10. Let W(n) and A(n) denote respectively, the worst case and average case running time of an algorithm executed on an input of size n. Which of the following is **ALWAYS TRUE?**

(a) $A(n) = \Omega(W(n))$ (b) $A(n) = \Theta(W(n))$

(c) $A(n) = O(W(n))$ (d) All of the above

11. Consider the following C-program fragment in which i, j and n are integer variables.

```
for (i = n, j = 0; i > 0;  i / = 2, j +=i);
```

Let val (j) denote the value stored in the variable j after termination of the for loop. Which one of the following is **true**?

(a) $val(j) = \Theta(\log n)$ (b) $val(j) = \Theta(\sqrt{n})$

(c) $val(j) = \Theta(n)$ (d) $val(j) = \Theta(n \log n)$

12. Consider the following segment of C-code:

```
int j, n;
j = 1;
while (j <= n)
j = j*2;
```

The number of comparisons made in the execution of the loop for any $n > 0$ is:

(a) $\lfloor \log_2 n + 2 \rfloor$

(b) n

(c) $\lfloor \log_2 n \rfloor$

(d) $\lfloor \log_2 n \rfloor + 1$

13. Consider the following function:

```
int unknown(int n)
{
   int i, j, k = 0;
   for (i = n/2;  i < = n;  i++)
   for(j = 2; j < = n; j = j*2)
   k = k + n/2;
   return (k);
}
```

The return value of the function is

(a) $\Theta(n^2)$ (b) $\Theta(n^2 \log n)$
(c) $\Theta(n^3)$ (d) $\Theta(n^3 \log n)$

14. The running time of the following algorithm

```
Procedure A(n)
If n<=2
return (1);
else
return (A[√n]);
```

Is best described by

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

15. Let A[1,…,n] be an array storing a bit (1 or 0) at each location, and f(m) is a function whose time complexity is o(m). Consider the following program fragment written in a C like language:

```
counter = 0;
for (i =1; i < = n; i++)
{
    if (A[i] = = 1) counter++;
    else
    {
        f (counter); counter = 0;
    }
}
```

The complexity of this program fragment is

(a) $\Omega(n^2)$ (b) $\Omega(n\log n)$ and $O(n^2)$
(c) $\Theta(n)$ (d) O(n)

16. The time complexity of the 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(nlog n)
(c) $O(n^2)$ (d) $O(2^n)$

17. The recurrence equation

$T(1) = 1$

$T(n) = 2T(n-1) + n, n \geq 2$

Evaluates to

(a) $2^{n+1} - n - 2$ (b) $2^n - n$
(c) $2^{n+1} - 2n - 2$ (d) $2^n + n$

18. What is the time complexity of the following recursive function?

```
int DoSomething (int n)
{
    if (n <= 2)
    return 1;
    else
    return (floor(sqrt(n)) + n);
}
```

(a) $\Theta(n^2)$ (b) $\Theta(n \log_2 n)$
(c) $\Theta(\log_2 n)$ (d) $\Theta(\log_2 \log_2 n)$

19. Solve the recurrence equations

$T(n) = T(n-1) + n$

$T(1) = 1$

20. What is the generating function G(z) for the sequence of Fibonacci numbers?

```
int f1 (int n)
{
   if (n==0| | n== 1)
            return n;
   else
            return (2 * f1( n–1)+ 3 * f1(n–2)) ;
}
int f2 (int n)
{
   int i;
   int X[N] , Y[N] , Z [N] ;
   X [0]= Y[0]= Z[0]=0;
   X [1]=1; Y [1]=2; Z[1]=3;
   for( i= 2; i<= n; i++)
   {
            X [i]= Y [i–1]+ Z[i–2] ;
            Y[i]= 2 * X [i];
            Z [i]= 3 * X[i] ;
   }
   return X[n] ;
}
```

The running time of f1 (n) and f2 (n) are

(a) $\Theta(n)$ and $\Theta(n)$

(b) $\Theta(2^n)$ and $\Theta(n)$

(c) $\Theta(n)$ and $\Theta(2^n)$

(d) $\Theta(2^n)$ and $\Theta(2^n)$

21. f1(8) and f2(8) return the values

(a) 1661 and 1640

(b) 59 and 59

(c) 1640 and 1640

(d) 1640 and 1661

02_Divide & Conquer.

Practice Questions

01. Merge sort uses
    (a) Divide and conquer strategy
    (b) Backtracking approach
    (c) Heuristic search
    (d) Greedy approach

02. For merging two sorted lists of sizes m and n into a sorted list of size m+n, we require comparisons of
    (a) O(m) (b) O(n)
    (c) O(m+n) (d) O(log m +log n)

03. If one uses straight two-way merge sort algorithm to sort the following elements in ascending order:

    20, 47, 15, 8, 9, 4, 40, 30, 12, 17

    Then the order of these elements after second pass of the algorithm is:
    (a) 8, 9, 15, 20, 47, 4, 12, 17, 30, 40
    (b) 8, 15, 20, 47, 4, 9, 30, 40, 12, 17
    (c) 15, 20, 47, 4, 8, 9, 12, 30, 40, 17
    (d) 4, 8, 9, 15, 20, 47, 12, 17, 30, 40

04. A list of n strings, each of length n, is sorted into lexicographic order using the merge-sort algorithm. The worst case running time of this computation is
    (a) O(n log n) (b) $O(n^2 \log n)$
    (c) $O(n^2 + \log n)$ (d) $O(n^2)$

05. You have an array of n elements. Suppose you implement quicksort by always choosing the central element of the array as the pivot. Then the tightest upper bound for the worst case performance is
    (a) $O(n^2)$ (b) $O(n^2 \log n)$
    (c) O(n log n) (d) $O(n^3)$

06. Let P be a quicksort program to sort numbers in ascending order. Let t_1 and t_2 be the time taken by the program for the inputs [1 2 3 4 5] and [5 4 3 2 1], respectively. Which of the following holds?
    (a) $t_1 = t_2$ (b) $t_1 > t_2$
    (c) $t_1 < t_2$ (d) $t_1 = t_2 + 5 \log 5$

07. Consider the Quick sort algorithm. Suppose there is a procedure for finding a pivot element which splits the list into two sub-lists each of which contains at least one-fifth of the elements. Let T(n) be the number of comparisons required to sort n elements. Then
    (a) $T(n) \le 2T(n/5) + n$
    (b) $T(n) \le T(n/5) + T(4n/5) + n$
    (c) $T(n) \le 2T(4n/5) + n$
    (d) $T(n) \le 2T(n/2) + n$

08. Randomized quick sort is an extension of quick sort where the pivot is chosen randomly. What is the worst case complexity of sorting n numbers using randomized quick sort?
    (a) O(n) (b) O(n log n)
    (c) $O(n^2)$ (d) O(n!)

09. The usual $\Theta(n^2)$ implementation of Insertion Sort to sort an array uses linear search to identify the position where an element is to be inserted into the already sorted part of the array. If, instead, we use binary search to identify the position, the worst case running time will
    (a) remain $\Theta(n^2)$
    (b) become $\Theta(n (\log n)^2)$
    (c) become $\Theta(n \log n)$
    (d) become $\Theta(n)$

10. Following algorithm (s) can be used to sort n integers in the range $[1\ldots n^3]$ in O(n) time?
(a) Heap sort (b) Quick sort
(c) Merge sort (d) Radix sort

11. An array of n numbers is given, where n is an even number. The maximum as well as the minimum of these n numbers needs to be determined. Which of the following is TRUE about the number of comparisons needed?
(a) At least 2n − c comparisons, for some constant c, are needed.
(b) At most 1.5n − 2 comparisons are needed.
(c) At least n log2 n comparisons are needed.
(d) O(n) .

12. The minimum number of comparisons required to find the minimum and the maximum of 100 numbers is ______.

13. The recurrence relation that arises in relation with the complexity of binary search is:
(a) T(n)= T(n/2) +k,k a constant
(b) T(n) = 2 T (n/2) +k,k a constant
(c) T(n) = T(n/2) + log n
(d) T(n) = T (n/2) + n

14. The solution to the recurrence equation
$T(2^k) = 3\,T(2^{k-1}) + 1$, T (1) = 1 is:
(a) 2^k (b) $\frac{(3^{k+1} - 1)}{2}$
(c) $3^{\log_2 k}$ (d) $2^{\log_3 k}$

15. Find a solution to the following recurrence equation
$$T(n) = T\left(\frac{n}{2}\right) + n$$
$$T(1) = 1$$

16. Which one of the following correctly determines the solution of the recurrence relation with T(1) = 1?
$$T(n) = 2T(n/2) + \log n$$
(a) $\Theta(n)$ (b) $\Theta(n \log n)$
(c) $\Theta(n^2)$ (d) $\Theta(\log n)$

17. The recurrence relation
$T(1) = 2$
$$T(n) = 3T\left(\frac{n}{4}\right) + n$$
Has the solution T(n) equal to
(a) O(n)
(b) O(log n)
(c) $O(n^{3/4})$
(d) None of the above

18. Let T(n) be the function defined by T(1) = 1, $T(n) = 2.\,T(n/2) + \sqrt{n}$ for $n \geq 2$
Which of the following statements is **true?**
(a)
(b) T(n) = O(n)
(c) T(n) = O(log n)
(d) None of the above

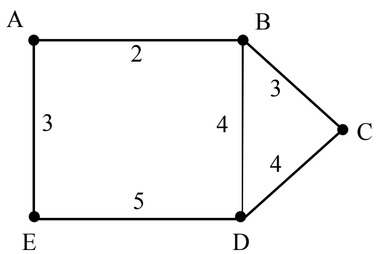
19. Suppose $T(n) = 2T\left(\frac{n}{2}\right) + n$
T (0) = T(1)
Which one of the following is **FALSE**?
(a) $T(n) = O(n^2)$
(b) $T(n) = \Theta(n \log n)$
(c) $T(n) = \Omega(n^2)$
(d) $T(n) = O(n \log n)$

20. The running time of an algorithm is represented by the following recurrence relation:
$$T(n) = \begin{cases} n & n \leq 3 \\ T\left(\frac{n}{3}\right) + cn & \text{otherwise} \end{cases}$$
Which one of the following represents the time complexity of the algorithm?
(a) $\Theta(n)$ (b) $\Theta(n \log n)$
(c) $\Theta(n^2)$ (d) $\Theta(n^2 \log n)$

03_Greedy Method.

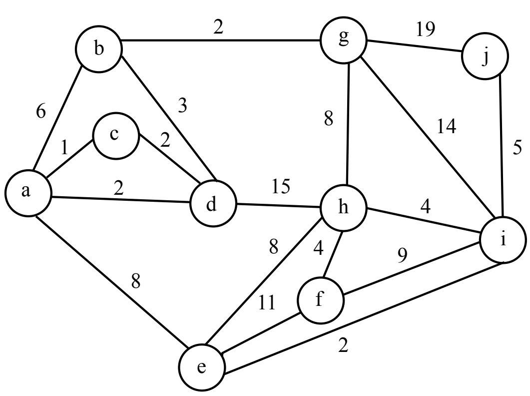
Practice Questions

01. How many minimum spanning trees does the following graph have? Draw them. (Weights are assigned to the edge).



02. Let G be an undirected connected graph with distinct edge weight. Let max e be the edge with maximum weight and min e be the edge with minimum weight. Which of the following statements is/are True?
    - (a) Every minimum spanning tree of G must contain min e
    - (b) If max e is in a minimum spanning tree, then its removal must disconnect G
    - (c) No minimum spanning tree contains max e
    - (d) G has a unique minimum spanning tree

03. What is the weight of a minimum spanning tree of the following graph?



(a) 29 (b) 31
(c) 38 (d) 41

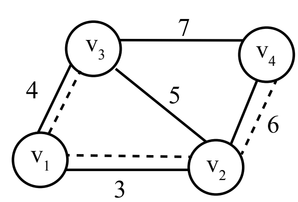
04. Consider a weighted complete graph G on the vertex set $\{v_1, v_2, \ldots, v_n\}$ such that the weight of, the edge (v_i, v_j) is $2\,|i-j|$. The weight of a minimum spanning tree of G is:

(a) n-1 (b) 2n-2
(c) n/2 (d) n^2

05. Let w be the minimum weight among all edge weights in an undirected connected graph. Let e be a specific edge of weight w. Which of the following is FALSE?
    - (a) No minimum spanning tree contains e.
    - (b) If e is not in a minimum spanning tree T, then the cycle formed by Adding e to T, all edges have the same weight.
    - (c) Every minimum spanning tree has an edge of weight w .
    - (d) e is present in every minimum spanning tree.

Common Data for Q06 & Q07.

An undirected graph G(V, E) contains $n(n > 2)$ nodes named $v_1, v_2, \ldots v_n$. Two nodes v_i, v_j are connected if and only if $0 < |i-j| \le 2$. Each edge (v_i, v_j) is assigned a weight i + j. A sample graph with n = 4 is shown below.



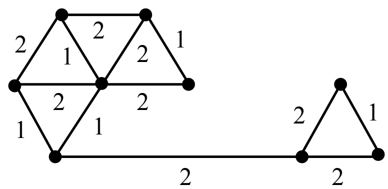
06. What will be the cost of the minimum spanning tree (MST) of such a graph with n nodes?

(a) $\frac{1}{12}(11n^2 - 5n)$ (b) $n^2 - n + 1$

(c) 6n - 11 (d) 2n + 1

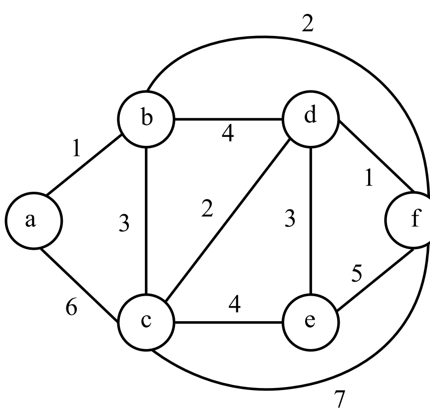
07. The length of the path from v_5 to v_6 in the MST of previous question with n=10 is
(a) 11
(b) 25
(c) 31
(d) 41

08. The number of distinct minimum spanning trees for the weighted graph below is ______.



09. Kruskal's algorithm for finding a minimum spanning tree of a weighted graph G with n vertices and m edges has the time complexity of:
(a) $O(n^2)$
(b) O(mn)
(c) O(m+n)
(d) O(m log n)

10. Consider the following graph:



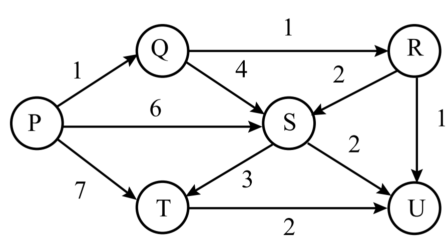
Which one of the following cannot be the sequence of edges added, in that order, to a minimum spanning tree using Kruskal's algorithm?
(a) (a-b), (d-f), (b-f), (d-c), (d-e)
(b) (a-b), (d-f), (d-c), (b-f), (d-e)
(c) (d-f), (a-b), (d-c), (b-f), (d-e)
(d) (d-f), (a-b), (b-f), (d-e), (d-c)

11. Let G = (V, E) be an undirected graph with a subgraph $G_1 = (V_1, E_1)$. Weights are assigned to edges of G as follows.

$$w(e) = \begin{cases} 0 & \text{if } e \in E_1 \\ 1 & \text{otherwise} \end{cases}$$

A single-source shortest path algorithm is executed on the weighted graph (V,E,w) with an arbitrary vertex v_1 of V_1 as the source. Which of the following can always be inferred from the path costs computed?
(a) The number of edges in the shortest paths from v_1 to all vertices of G
(b) G_1 is connected
(c) V_1 forms a clique in G
(d) G_1 is a tree

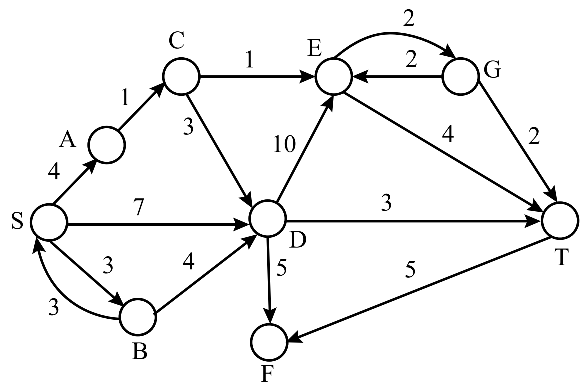
12. Suppose we run Dijkstra's single source shortest-path algorithm on the following edge-weighted directed graph with vertex P as the source.



In what order do the nodes get included into the set of vertices for which the shortest path distances are finalized?
(a) P, Q, R, S, T, U
(b) P, Q, R, U, S, T
(c) P, Q, R, U, T, S
(d) P, Q, T, R, U, S

13. To implement Dijkstra's shortest path algorithm on weighted graphs so that it runs in linear time, the data structure to be used is:
(a) Queue
(b) Stack
(c) Heap
(d) B-Tree

14. Consider the directed graph shown in the figure below. There are multiple shortest paths between vertices S and T. Which one will be reported by Dijkstra's shortest path algorithm? Assume that, in any iteration, the shortest path to a vertex v is updated only when a strictly shorter path to v is discovered.



(a) SDT (b) SBDT
(c) SACDT (d) SACET

Data for Question No.15.

A Language uses an alphabet of six letters, {a,b,c,d,e,f}. The relative frequency to use of each letter of the alphabet in the language is as given below.

Letter	Relative frequency of use
a	0.19
b	0.05
c	0.17
d	0.08
e	0.40
f	0.11

Design a prefix binary code for the language which would minimize the average length of the encoded words of the language.

Common Data for Q16 & Q17.

Suppose the letters a, b, c, d, e, f have probabilities

$\frac{1}{2}, \frac{1}{4}, \frac{1}{8}, \frac{1}{16}, \frac{1}{32}, \frac{1}{32}$ respectively.

16. Which of the following is the Huffman code for the letter a, b, c, d, e, f?
(a) 0, 10, 110, 1110, 11110, 11111
(b) 11, 10, 011, 010, 001, 000
(c) 11, 10, 01, 001, 0001, 0000
(d) 110, 100, 010, 000, 001, 111

17. What is the average length of the correct answer to Q.16?
(a) 3 (b) 2.1875
(c) 2.25 (d) 1.9375

04_Graph Techniques,Components, Heaps.

Practice Questions

01. The correct matching for the following pairs is

List – I

(P) All pairs shortest paths

(Q) Quick Sort

(R) Minimum weight spanning tree

(S) Connected Components

List - II

1. Greedy
2. Depth-First search
3. Dynamic Programming
4. Divide and Conquer

(a) P – 2; Q – 4; R – 1; S - 3

(b) P – 3; Q – 4; R – 1; S - 2

(c) P – 3; Q – 4; R – 2; S - 1

(d) P – 4; Q – 1; R – 2; S – 3

02. Give the correct matching for the following pairs:

(1) O(log n)	(P) Selection sort
(2) O(n)	(Q) Insertion sort
(3) O(n log n)	(R) Binary search
(4) $O(n^2)$	(S) Merge sort

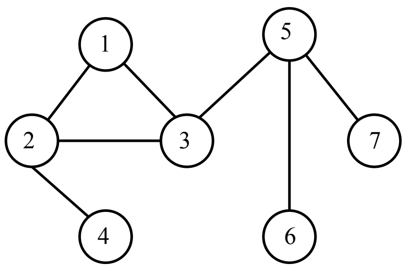
(a) 1 – R; 2 – P; 3 – Q; 4 - S

(b) 1 – R; 2 – Q; 3– S; 4 - P

(c) 1 – P; 2– R; 3 – S; 4 - Q

(d) 1 – P; 2 – S; 3 – R; 4 – Q

03. The number of articulation points of the following graph is



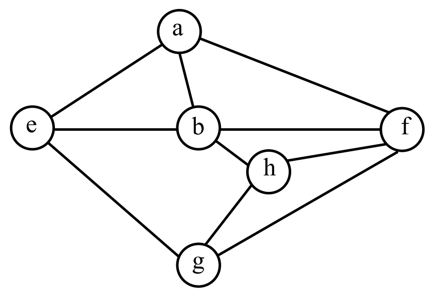
(a) 0 (b) 1 (c) 2 (d) 3

04. Consider any array representation of an n element binary heap where the elements are stored from index 1 to index n of the array. For the element stored at index i of the array (i<=n), the index of the parent is

(a) i-1 (b) $\lfloor i/2 \rfloor$

(c) $\lceil i/2 \rceil$ (d) (i+1)/2

05. Consider the following graph



Among the following sequences

(a) a b e g h f

(b) a b f e h g

(c) a b f h g e

(d) a f g h b e

Which are depth first traversals of the above graph?

06. In a heap with n elements with the smallest element at the root, the 7^{th} smallest element can be found in time

(a) $\Theta(n \log n)$ (b) $\Theta(n)$

(c) $\Theta(\log n)$ (d) $\Theta(1)$

07. The tightest lower bound on the number of comparisons, in the worst case, for comparison-based sorting is of the order of

(a) n (b) n^2

(c) nlogn (d) $n\log^2 n$

08. In a binary max heap containing n numbers, the smallest element can be found in time
(a) $\Theta(n)$ (b) $\Theta(\log n)$
(c) $\Theta(\log \log n)$ (d) $\Theta(1)$

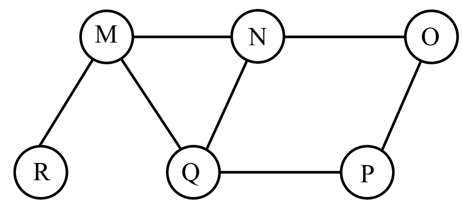
09. Which one of the following in-place sorting algorithms needs the minimum number of swaps?
(a) Quick sort
(b) Insertion sort
(c) Selection sort
(d) Heap sort

10. An element in an array X is called a leader if it is greater than all elements to the right of it in X. The best algorithm to find all leaders in an array
(a) Solves it in linear time using a left to right pass of the array
(b) Solves it in linear time using a right to left pass of the array
(c) Solves it using divide and conquer in time $\Theta(n \log n)$
(d) Solves it in time $\Theta(n^2)$

11. Which of the following sorting algorithms has the lowest worst-case complexity?
(a) Merge sort (b) Bubble sort
(c) Quick sort (d) Selection sort

12. The most efficient algorithm for finding the number of connected components in an undirected graph on n vertices and m edges has time complexity
(a) $\Theta(n)$ (b) $\Theta(m)$
(c) $\Theta(m+n)$ (d) $\Theta(mn)$

13. The Breadth First Search algorithm has been implemented using the queue data structure. Which of the following is NOT order of visiting the nodes of the following graph is



(a) MNOPQR (b) NQMPOR
(c) QMNPRO (d) QMNPOR

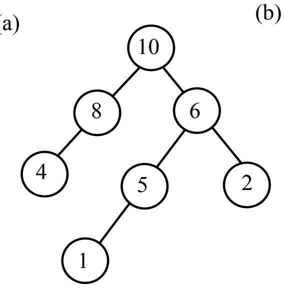
14. What is the number of swaps required to sort n elements using selection sort, in the worst case?
(a) $\Theta(n)$ (b) $\Theta(n \log n)$
(c) $\Theta(n^2)$ (d) $\Theta(n^2 \log n)$

15. Which one of the following array represents a binary max-heap?
(a) {25,12,16,13,10,8,14}
(b) {25,14,13,16,10,8,12}
(c) {25,14,16,13,10,8,12}
(d) {25,14,12,13,10,8,16}

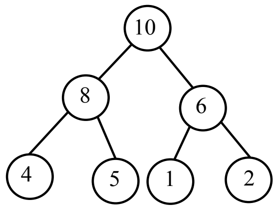
16. What is the content of the array after two delete operations on the correct answer to the previous question?
(a) {14,13,12,10,8}
(b) {14,12,13,8,10}
(c) {14,13,8,12,10}
(d) {14,13,12,8,10}

17. A max-heap is a heap where the value of each parent is greater than or equal to the value of its children. Which of the following is a max-heap?

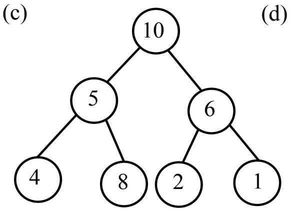
(a)



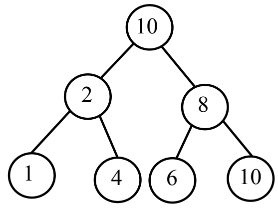
(b)



(c)



(d)



18. Match the pairs in the following:

List - I

(a) Straseen's matrix multiplication algorithm
(b) Kruskal's minimum spanning tree algorithm
(c) Biconnected components algorithm
(d) Floyd's shortest path algorithm

List – II

(p) Greedy method
(q) Dynamic programming
(r) Divide and Conquer
(s) Depth first search

19. Match the pairs in the following

List - I

(a) Heap construction
(b) Constructing hash table with linear probing
(c) AVL Tree construction
(d) Digital tree construction

List - II

(p) Ω(n log10 n)
(q) O(n)
(r) O(n^2)
(s) O(n $\log_2$ n)

20. Following algorithm(s) can be used to sort n integers in the range [1….n^3] in O(n) time
(a) Heap sort (b) Quick sort
(c) Merge sort (d) Radix sort

21. The minimum number of interchanges needed to convert the array

89, 19, 40, 17, 12, 10, 2, 5, 7, 11, 6, 9, 70

into a heap with the maximum element at the root is
(a) 0 (b) 1 (c) 2 (d) 3

22. The average number of key comparisons done on a successful sequential search in list of length n is
(a) log n (b) $\frac{n-1}{2}$ (c) n/2 (d) $\frac{n+1}{2}$

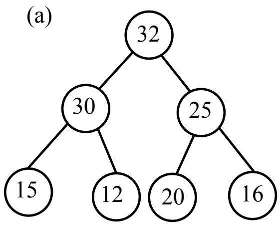
23. Let G be an undirected graph. Consider a depth-first traversal of G, and let T be the resulting depth-first search tree. Let u be a vertex in G and let v be the first new (unvisited) vertex visited after visiting u in the traversal. Which of the following statements is always **true**?
(a) {u,v} must be an edge in G, and u is a descendant of v in T
(b) {u,v} must be an edge in G, and v is a descendant of u in T
(c) If {u,v} is not an edge in G then u is a leaf in T
(d) If {u,v} is not an edge in G then u and v must have the same parent in T

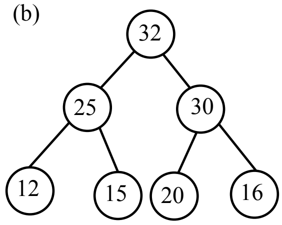
24. Consider an undirected unweighted graph G. Let a breadth-first traversal of G be done starting from a node r. Let d(r,u) and d(r,v) be the lengths of the shortest paths from r to u and v respectively in G. If u is visited before v during the breadth-first traversal, which of the following statements is **correct**?

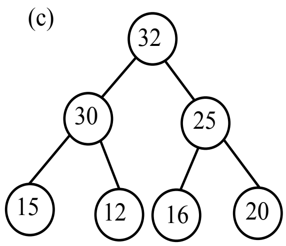
(a) $d(r,u)<d(r,v)$ (b) $d(r,u)>d(r,v)$
(c) $d(r,u) \leq d(r,v)$ (d) None of the above

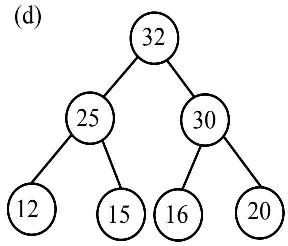
25. How many undirected graphs (not necessarily connected) can be constructed out of a given set $V = \{v_1, v_2,\ldots,v_n\}$ of n vertices?
(a) $n(n-1)/2$ (b) 2^n
(c) $n!$ (d) $2^{n(n-1)/2}$

26. The elements 32, 15, 20, 30, 12, 25, 16, are inserted one by one in the given order into a max Heap. The resultant max Heap is

(a)


(b)


(c)


(d)


27. Let T be a depth first search tree in an undirected graph G. Vertices u and v are leaves of this tree T. The degrees of both u and v in G are at least 2. Which one of the following statements is **true**?
(a) There must exist a vertex w adjacent to both u and v in G
(b) There must exist a vertex w whose removal disconnects u and v in G
(c) There must exist a cycle in G containing u and v
(d) There must exist a cycle in G containing u and all its neighbors in G.

Common Data for Questions 28 & 29

A 3-ary max heap is like a binary max heap, but instead of 2 children, nodes have 3 children. A 3-ary heap can be represented by an array as follows: The root is stored in the first location, a[0], nodes in the next level, from left to right, is stored from a[1] to a[3]. The nodes from the second level of the tree from the left to right are stored from a[4] location onward. An item x can be inserted into a 3-ary heap containing n items by placing x in the location a[n] and pushing it up the tree to satisfy the heap property.

28. Which one of the following is a valid sequence of elements in an array representing 3-ary max heap?
(a) 1, 3, 5, 6, 8, 9 (b) 9, 6, 3, 1, 8, 5
(c) 9, 3, 6, 8, 5, 1 (d) 9, 5, 6, 8, 3, 1

29. Suppose the elements 7, 2, 10 and 4 are inserted, in that order, into the valid 3-ary max heap found in the ***above question***. Which one of the following is the sequence of items in the array representing the resultant heap?
(a) 10, 7, 9, 8, 3, 1, 5, 2, 6, 4
(b) 10, 9, 8, 7, 6, 5, 4, 3, 2, 1
(c) 10, 9, 4, 5, 7, 6, 8, 2, 1, 3
(d) 10, 8, 6, 9, 7, 2, 3, 4, 1, 5

30. Consider the process of inserting an element into a Max Heap, where the Max Heap is represented by an array. Suppose we perform a binary search on the path from the new leaf to the root to find the position for the newly inserted element, the number of comparisons performed is:
(a) $\Theta(\log_2 n)$ (b) $\Theta(\log_2 \log_2 n)$
(c) $\Theta(n)$ (d) $\Theta(n \log_2 n)$

31. We have a binary heap on n elements and wish to insert n more elements (not necessarily one after another) into this heap. The total time required for this is

(a) $\Theta(\log n)$ (b) $\Theta(n)$
(c) $O(n \log n)$ (d) $\Theta(n^2)$

05_Dynamic Programming.

Practice Questions

All Pairs shortest Path

01. Which one of the following algorithm design techniques is used in solving Sum of Subsets problem?
(a) Dynamic programming
(b) Backtracking
(c) Greedy
(d) Divide and Conquer

02. Fill in the blanks in the following template of an algorithm to compute all pairs shortest path lengths in a directed graph G with n*n adjacency matrix A. A[i,j] equals to 1 if there is an edge in G from i to j, and 0 otherwise. Your aim in filling in the blanks is to ensure that the algorithm is **correct**?

```
INITIALIZATION: For i = 1 … n
{
   for j = 1 … n
   {
      if A[i,j]=0 then P[i,j] = ____
      else  P[i,j] =____;
   }
ALGORITHM: For i = 1 …n
{
   for j = 1 …n
   {
     for k = 1 …n
     {
      P[__,__]= min{_______,_______};
     }
   }
}
```

(a) Copy the complete line containing the blanks in the Initialization step and fill in the blanks.
(b) Copy the complete line containing the blanks in the Algorithm step and fill in the blanks.
(c) Fill in the blank: The running time of the Algorithm is O (____).

Longest Common Subsequence (LCS)

Common Data for Questions 03 & 04

A sub-sequence of a given sequence is just the given sequence with some elements (possibly none or all) left out. We are given two sequences X[m] and Y[n] of lengths m and n, respectively, with indexes of X and Y starting from 0

03. We wish to find the length of the longest common sub-sequence (LCS) of X[m] and Y[n] as l(m,n), where an incomplete recursive definition for the function l(i,j) to compute the length of the LCS of X[m] and Y[n] is given below:

l (i ,j) = 0, if either i= 0 or j=0

= expr1, if i,j > 0 and

x[i-1] = y [j-1]

= expr2, if i,j>0 and

x[i-1] ≠ y[j-1]

Which one of the following options is **correct**?

(a) expr1≡ l(i-1,j) +1

(b) expr1≡ l(i,j-1)

(c) expr2≡ max (l(i-1,j), l(i,j-1))

(d) expr2 ≡ max (l(i-1,j-1), l(i,j))

04. The values of $l(i, j)$ could be obtained by dynamic programming based on the correct recursive definition of $l(i, j)$ of the form given above, using an array L[M,N], where M= m+1 and N = n+1, such that L[i, j] = $l(i, j)$.

Which one of the following statements would be **TRUE** regarding the dynamic programming solution for the recursive definition of $l(i, j)$?

(a) All elements of L should be initialized to 0 for the values of $l(i, j)$ to be Properly computed.

(b) The values of $l(i, j)$ may be computed in a row major order or column Major order of L(M,N).

(c) The values of $l(i, j)$ cannot be computed in either row major order or Column major order of L(M,N).

(d) L[p,q] needs to be computed before L[r,s] if either p<r or q<s.

05. Consider two strings A = "*qpqrr*" and B = "*pqprqrp*". Let x be the length of the longest common subsequence (not necessarily contiguous) between A and B and let *y* be the number of such longest common subsequences between A and B. Then x + 10*y* = ____.

Statement for Linked Answer Questions 06 & 07

The subset-sum problem is defined as follows. Given a set of n positive integers, S = $\{a_1, a_2, a_3, \ldots, a_n\}$, and positive integer W, is there a subset of S whose elements sum to W? A dynamic program for solving this problem uses a 2-dimensional Boolean array, X, with n rows and W+1 columns X[i, j], $1 \le i \le n, 0 \le j \le W$, is **TRUE** if and only if there is a subset of $\{a_1, a_2, a_3, \ldots, a_n\}$ whose elements sum to j.

06. Which of the following is valid for $2 \le i \le n$ and $a_i \le j \le W$?

(a) $X[i, j] = X[i-1, j] \vee X[i, j-a_i]$

(b) $X[i, j] = X[i-1, j] \vee X[i-1, j-a_i]$

(c) $X[i, j] = X[i-1, j] \wedge X[i, j-a_i]$

(d) $X[i, j] = X[i-1, j] \wedge X[i-1, j-a_i]$

07. Which entry of the array X, if **TRUE**, implies that there is a subset whose elements sum to W?

(a) X[1, w] (b) X[n, 0]

(c) X[n, W] (d) X[n – 1, n]

Matrix Chain Multiplication

08. Four matrices M_1, M_2, M_3 and M_4 are dimensions p × q, q × r, r × s and s × t respectively can be multiplied in several ways with different number of total scalar multiplications. For example When multiplied as $((M_1 \times M_2) \times (M_3 \times M_4))$ total number of scalar multiplications is pqr+rst+prt. When multiplied as $(((M_1 \times M_2) \times M_3) \times M_4)$ the total number of scalar multiplications is pqr+prs+ pst. If p=10, q=100, r=20, s=5 and t=80, then the minimum number of scalar multiplications needed is

(a) 248000 (b) 44000

(c) 19000 (d) 25000