



Ashima Garg

Course: GATE  
 Computer Science Engineering(CS)

HOME

MY TEST

BOOKMARKS

MY PROFILE

REPORTS

BUY PACKAGE

ASK AN EXPERT

OFFER

EXCLUSIVE OFFER FOR OTS STUDENTS ONLY ON BOOK PACKAGES

## SINGLE SUBJECT :ALGORITHMS (GATE - 2019) - REPORTS

[OVERALL ANALYSIS](#)
[COMPARISON REPORT](#)
[SOLUTION REPORT](#)

ALL(33)

CORRECT(19)

INCORRECT(12)

SKIPPED(2)

### Q. 1

What data structure would you choose to implement “undo” feature in a word processor?

[Have any Doubt ?](#)

A

Queue

B

Dequeue

C

Stack

**Your answer is Correct**
**Solution :**

(c)

Stack is most suitable data structure since it works in LIFO ordering. All the actions are pushed into stack and whatever action has to be reversed (undone) can be just popped off the top of stack.

D

None of the above

[QUESTION ANALYTICS](#)

### Q. 2

Which of the following tree traversal technique is not based on depth first search?

[Have any Doubt ?](#)

A

Preorder traversal

B

Level order traversal

**Your answer is Correct**
**Solution :**

(b)

Level order traversal also known as breadth first search while other three are depth first search.

C

Postorder traversal

D

Inorder traversal

[QUESTION ANALYTICS](#)

### Q. 3

Consider the following statement:

C : Ternary search is more efficient than binary search for finding an element in a sorted array.



Ashima Garg

 Course: GATE  
 Computer Science Engineering(CS)

HOME

MY TEST

BOOKMARKS

MY PROFILE

REPORTS

BUY PACKAGE

ASK AN EXPERT

OFFER

 EXCLUSIVE OFFER FOR OTS  
 STUDENTS ONLY ON BOOK  
 PACKAGES

A

S<sub>1</sub> is correct, S<sub>2</sub> is not

B

S<sub>2</sub> is correct, S<sub>1</sub> is not

Your answer is Correct

**Solution :**

(b)

- Recurrence relation for ternary search:  $T(n) = T(n/3) + 4$ ,  $T(1) = 1$
- Recurrence relation for binary search:  $T(n) = T(n/2) + 2$ ,  $T(1) = 1$

**Number of comparisons**Binary Search:  $2\log_2 n + 1$ Ternary Search:  $4\log_3 n + 1$ Since  $(2 \log_2 n + 1) < (4 \log_3 n + 1)$ . Hence binary search is preferred.

- In AVL tree, worst case deletion and insertion both have  $O(\log n)$  complexity

C

Both are correct

D

Both are incorrect

QUESTION ANALYTICS

**Q. 4**

Match List-I (Solution approach) with List-II (Problems) and select the correct answer using codes given below the lists:

**List-I**

- A. Divide and Conquer Approach  
 B. Dynamic programming  
 C. Greedy Approach  
 D. Backtracking

**Codes:**

	A	B	C	D
(a)	3	1	2	4
(b)	3	2	1	4
(c)	3	2	4	1
(d)	4	3	1	2

**List-II**

1. Graph coloring
2. 0/1 Knapsack
3. Strassen Matrix Multiplication
4. Huffman Encoding

Have any Doubt ?

A

a

B

b

C

c

Your answer is Correct

**Solution :**

(c)

- Divide and Conquer Approach - Strassen Matrix Multiplication  
 Dynamic programming - 0/1 Knapsack  
 Greedy Approach - Huffman Encoding  
 Backtracking - Graph coloring

D

d



Ashima Garg

 Course: GATE  
 Computer Science Engineering(CS)

HOME

MY TEST

BOOKMARKS

MY PROFILE

REPORTS

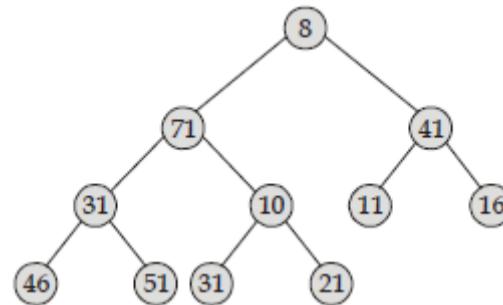
BUY PACKAGE

ASK AN EXPERT

OFFER

 EXCLUSIVE OFFER FOR OTS  
 STUDENTS ONLY ON BOOK  
 PACKAGES
**Q. 5**

A min-max heap has the following property: "A almost complete binary tree where each node at an even level in the tree is less than all of its descendants and each of the node at an odd level in the tree is greater than all of its descendants".



What is the worst case time complexity of "deleting minimum element" and "deleting maximum element" on above data structure?

A

 $O(\log n)$  and  $O(\log n)$ 

Correct Option

**Solution :**

(a)

`deletemin()`: Root element is the smallest element. Remove it and swap it with the rightmost leaf node to maintain heap property. Then search for the minimum element out of the remaining elements by looking in the lowermost min (even) level. Let it be at index ' $m$ '. Swap this element at index  $m$  with the root. Do this recursively for the subtree rooted at index  $m$  to maintain the min-max property.  
`deletemax()`: The max element is one out of the two children of root. Find it and then follow the same steps as `deletemin()` operation, this time choosing the max element out of the remaining to be swapped at every stage.

B

 $O(n)$  and  $O(\log n)$ 

C

 $O(\log n)$  and  $O(n)$ 

D

 $O(n)$  and  $O(n)$ 

Your answer is Wrong

QUESTION ANALYTICS

**Q. 6**

Consider the recurrence relation:

$$T(n) = \begin{cases} T(\sqrt{n}) + \log n & \text{if } n \geq 2 \\ O(1) & \text{else} \end{cases}$$

Then  $T(n)$  in terms of  $\Theta$  notation is

A

 $\Theta(\log n)$ 

Your answer is Correct

**Solution :**

(a)

$$T(n) = T(\sqrt{n}) + \log n$$

$$\text{Put, } n = 2^m$$

$$T(2^m) = T(2^{m/2}) + m$$

$$\text{Put, } T(2) = S($$

$$\text{So, } S(m) = S\left(\frac{m}{2}\right) + m$$

By applying Master theorem:

We get,  $S(m) = \Theta(m)$



Ashima Garg

 Course: GATE  
 Computer Science Engineering(CS)

HOME

MY TEST

BOOKMARKS

MY PROFILE

REPORTS

BUY PACKAGE

ASK AN EXPERT

OFFER

 EXCLUSIVE OFFER FOR OTS  
 STUDENTS ONLY ON BOOK  
 PACKAGES

 B  
 $\Theta(n)$ 

 C  
 $\Theta(\log n)$ 

 D  
 $\Theta(n \log n)$ 

## QUESTION ANALYTICS

## Q. 7

Consider we have given two large integers in binary format each of length  $n$  bits. What is the worst case time complexity to find product of two integers?

Have any Doubt ?

 A  
 $O(\log n)$ 

 B  
 $O(n)$ 

Your answer is Wrong

 C  
 $O(n^{1.59})$ 

Correct Option

## Solution :

(c)

The algorithm is known as Karatsuba algorithm and is based on divide and conquer strategy.  
 Recurrence relation is

$$T(n) = 3T\left(\frac{n}{2}\right) + O(n)$$

Solution of this recurrence is  $O(n^{1.59})$ .

 D  
 $O(n^2)$ 

## QUESTION ANALYTICS

## Q. 8

Which of the following is correct?

Have any Doubt ?

 A  

$$\frac{e^{n \log n}}{n} < 2^{n \log n} < n^{\sqrt{n}}$$

 B  

$$2^{n \log n} < \frac{e^{n \log n}}{n} < n^{\sqrt{n}}$$

 C  

$$n^{\sqrt{n}} < \frac{e^{n \log n}}{n} < 2^{n \log n}$$

Your answer is Correct



Ashima Garg

 Course: GATE  
 Computer Science Engineering(CS)

HOME

MY TEST

BOOKMARKS

MY PROFILE

REPORTS

BUY PACKAGE

ASK AN EXPERT

OFFER

 EXCLUSIVE OFFER FOR OTS  
 STUDENTS ONLY ON BOOK  
 PACKAGES

2.

$$\begin{aligned} 2^{n \log n} &= e^{(\log_2)^{n \log n}} \\ &= (e^{n \log n}) \end{aligned}$$

3.

$$\begin{aligned} n^{\sqrt{n}} &= (e^{\log n})^{\sqrt{n}} \\ &= (e^{\sqrt{n} \log n}) \end{aligned}$$

D

$$2^{n \log n} < n^{\sqrt{n}} < \frac{e^{n \log n}}{n}$$

QUESTION ANALYTICS

**Q. 9**

What is the time complexity of the following program?

```
void fun(int n) {
    int i, j, counter = 0;
    for (i = 1; i ≤ n; i++) {
        for (j = 1; j × j ≤ n; j++)
            count++;
    }
}
```

Have any Doubt?

A

 $O(n \log n)$ 

B

 $O(n^2)$ 

C

 $O(n \log \log n)$ 

D

 $O(n^{3/2})$ 

Your answer is Correct

**Solution :**

(d)

```
for (i = 1; i ≤ n; i++) ⇒ O(n)
for (j = 1; j × j ≤ n; j++)
    count++;
Total time complexity = O(n × n^{1/2})
= O(n^{3/2})
```

QUESTION ANALYTICS

**Q. 10**

Consider you are playing game of shooting balloon and you are expected to shoots ' $n$ ' balloon in the board. If you are a sharp shooter (100% accuracy) and for every 2 balloons you are able to shoot, one new balloon is inserted in the board then what is the time complexity of this shooting procedure if the board has to be emptied?

FAQ | Have any Doubt?

A

 $O(1)$



Ashima Garg

 Course: GATE  
 Computer Science Engineering(CS)

HOME

MY TEST

BOOKMARKS

MY PROFILE

REPORTS

BUY PACKAGE

ASK AN EXPERT

OFFER

 EXCLUSIVE OFFER FOR OTS  
 STUDENTS ONLY ON BOOK  
 PACKAGES
**Solution :**

(b)

Number of balloons originally	Total balloons shot
1	1
2	3
3	5
$n$	$2 \times n - 1$

C

 $O(\log n)$ 

D

 $O(n^2)$ 

QUESTION ANALYTICS

**Q. 11**
 The minimum number of comparisons to find the biggest element in an unsorted list of size 15 is \_\_\_\_\_.  
[Have any Doubt ?](#)

14

Your answer is **Correct** 14**Solution :**

(14)

 $(n - 1)$  comparisons are required to find the biggest element in an unsorted list of size  $n$ .

QUESTION ANALYTICS

**Q. 12**

Consider the following program given below along with the input X[] and Y[]:

```

int Sequence(char *X, char *Y) {
    int m = strlen(X), n = strlen(Y);
    int l = LCS(X, Y, m, n);
    return (m + n - l);
}
int LCS( char *X, char *Y, int m, int n )
{
    int L[m + 1][n + 1];
    int i, j;
    for (i = 0; i ≤ m; i++) {
        for (j = 0; j ≤ n; j++) {
            if (i == 0 || j == 0)
                L[i][j] = 0;
            else if (X[i - 1] == Y[j - 1])
                L[i][j] = L[i - 1][j - 1] + 1;
            else
                L[i][j] = max(L[i - 1][j], L[i][j - 1]);
        }
    }
    return L[m][n];
}
int main( ) {
    char X[ ] = "MADEEASY"; char Y[ ] = "EASYWAVES";
    printf("%d\n", Sequence(X, Y));
    return 0;
}
  
```



Ashima Garg

 Course: GATE  
 Computer Science Engineering(CS)

HOME

MY TEST

BOOKMARKS

MY PROFILE

REPORTS

BUY PACKAGE

ASK AN EXPERT

OFFER

 EXCLUSIVE OFFER FOR OTS  
 STUDENTS ONLY ON BOOK  
 PACKAGES

12

Your answer is Correct12

**Solution :**

(12)

In the above program 1<sup>st</sup> we have to find the longest common subsequence which is 'EASY', function Sequence( ) will gives the length of minimum length string which contain LCS of both string i.e. length (MADEEASYSWAYS) = 12.

QUESTION ANALYTICS

**Q. 13**

Given a hash table with load factor of 300 that stores 10800 elements, the number of slots are \_\_\_\_\_.

Have any Doubt ?

36

Your answer is Correct36

**Solution :**

(36)

$$\text{Load factor} = \frac{\text{Number of elements}}{\text{Number of slots}}$$

$$300 = \frac{10800}{x}$$

$$x = 36$$

QUESTION ANALYTICS

**Q. 14**
 Consider the following scenario during insertion sort when the array looks like the following:  
 {25, 75, 95, 125, 80, 5, 10}

The number of comparisons that it will further take for the array to be completely sorted are \_\_\_\_\_.

Have any Doubt ?

17

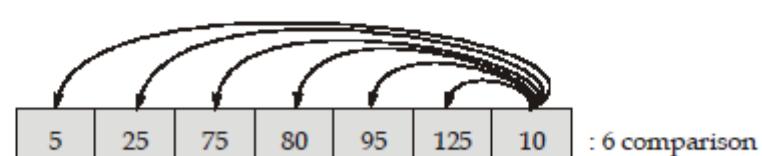
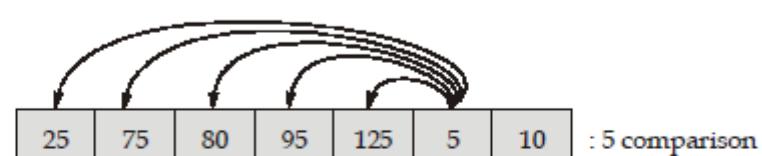
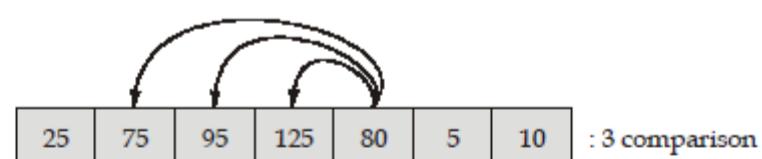
Correct Option

**Solution :**

(17)

$$17 = 3 + 5 + 6 + 3$$

3 comparison, one each for 75, 95, 125





Ashima Garg

 Course: GATE  
 Computer Science Engineering(CS)

HOME

MY TEST

BOOKMARKS

MY PROFILE

REPORTS

BUY PACKAGE

ASK AN EXPERT

OFFER

 EXCLUSIVE OFFER FOR OTS  
 STUDENTS ONLY ON BOOK  
 PACKAGES

## QUESTION ANALYTICS

Your Answer is 14

**Q. 15**

Consider two strings  $x = "bcde"$  and  $y = "cdgf"$ , with following operations to convert  $x$  to  $y$ :  
 Insert a character into  $x$  (at any position).  
 Delete a character from  $x$  (from any position).  
 Replace a character in  $x$  by another character.  
 The minimum number of such operation required are \_\_\_\_\_.

[Have any Doubt ?](#)

3

Correct Option

**Solution :**

13

1. Delete the beginning  $b$  from  $x$ .  $x$  becomes  $cde$ .
2. Replace character  $e$  in  $x$  by character  $g$ .  $x$  becomes  $cdg$ .
3. Insert character  $f$  at the end of  $x$ .  $x$  becomes  $cdgf$ .

## QUESTION ANALYTICS

Your Answer is 4

**Q. 16**

A professor gave Ramesh and Suresh task to compute  $nm$ . The algorithm used by Ramesh and Suresh are given below:

**Ramesh**

```
int power1(int x, int y)
{
    int temp;
    if (y == 0) return 1;
    temp = power1(x, y/2);
    if (y%2 == 0)
        return temp * temp;
    else
        return x * temp * temp;
}
int main ()
{
    power1(a, b);
}
```

**Suresh**

```
int power2(int x, int y)
{
    if (y == 0) return 1;
    else if (y%2 == 0)
        return power2(x, y/2) * power2(x, y/2);
    else
        return x * power2(x, y/2) * power2(x, y/2);
}
int main ()
{
    power2(a, b);
}
```

The difference in time taken by Ramesh and Suresh algorithm to compute  $44^{256}$  is \_\_\_\_\_.

[Have any Doubt ?](#)

248

Correct Option

**Solution :**

(248)

Complexity of Ramesh algorithm =  $\log m$ Complexity of Suresh algorithm =  $m$ 

$$|8 - 256| = |248|$$



Ashima Garg

 Course: GATE  
 Computer Science Engineering(CS)

HOME

MY TEST

BOOKMARKS

MY PROFILE

REPORTS

BUY PACKAGE

ASK AN EXPERT

OFFER

 EXCLUSIVE OFFER FOR OTS  
STUDENTS ONLY ON BOOK  
PACKAGES
**Q. 17**

The preorder traversal of a Binary Search Tree is 100, 50, 20, 30, 60, 55, 150, 130, 140, 180, 170. What would be its postorder traversal?

[Have any Doubt ?](#)

A

20, 30, 55, 60, 50, 130, 140, 180, 170, 150, 100

B

20, 30, 55, 60, 50, 140, 130, 180, 170, 150, 100

C

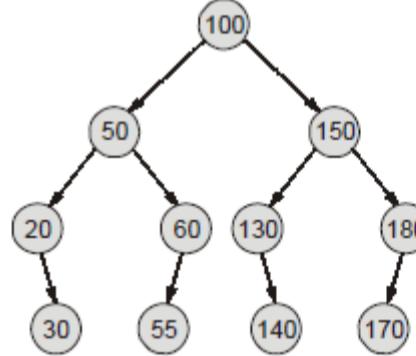
30, 20, 55, 60, 50, 140, 130, 170, 180, 150, 100

Your answer is Correct

**Solution :**

(c)

Inorder traversal of Binary Search Tree is in sorted order so,  
 Inorder traversal: 20, 30, 50, 55, 60, 100, 130, 140, 150, 170, 180  
 With inorder and preorder, unique Binary Search Tree can be determined.



Postorder traversal will be: 30, 20, 55, 60, 50, 140, 130, 170, 180, 150, 100

D

30, 20, 60, 55, 50, 130, 140, 180, 170, 150, 100

QUESTION ANALYTICS

**Q. 18**

Assume an array of  $n$  integers and a function Max( ) given below:

Max( ) = {Find an element which is higher than its left and right element if any}

What is the worst case time complexity for Max( ) on an input array?

[Have any Doubt ?](#)

A

 $O(\log n)$ 

B

 $O(n)$ 

Your answer is Correct

**Solution :**

(b)

Traverse the array element wise and for each element, compare its value with its left adjacent and right adjacent element (if it exists). If the condition is satisfied, print the elements.

C

 $O(n \log n)$ 

D

 $O(n^2)$



Ashima Garg

 Course: GATE  
 Computer Science Engineering(CS)

HOME

MY TEST

BOOKMARKS

MY PROFILE

REPORTS

BUY PACKAGE

ASK AN EXPERT

OFFER

 EXCLUSIVE OFFER FOR OTS  
 STUDENTS ONLY ON BOOK  
 PACKAGES
**Q. 19**

Given a degree sequence for ' $n$ ' vertices i.e.,  $d_1, d_2, \dots, d_n$ . What is the worst case time complexity to determine if a simple graph is possible with the given degree sequence?

[Have any Doubt ?](#)

A

 $O(n)$ 

B

 $O(n^2 \log n)$ 

Correct Option

**Solution :**

(b)

Since a degree sequence is given, Havale-Hakimi algorithm can be applied to it. In this initially we have to sort the given degree sequence, and then decrement the degree sequence  $a[i]^{\text{th}}$  time  $i$  starting from 0. The obtained output is sorted again. In this way, the algorithm will take  $O(n^2 \log n)$  time.

C

 $O(n \log n)$ 

D

 $O(n^2)$ 

Your answer is Wrong

QUESTION ANALYTICS

**Q. 20**

What is the worst case time complexity to delete middle element from min heap of  $n$  distinct elements?

[Have any Doubt ?](#)

A

 $O(\log n)$ 

Correct Option

**Solution :**

(a)

We can find the middle element of the heap, by simply access is the  $n/2^{\text{th}}$  element of the array which can be done in  $O(1)$  time. Further that element is replaced by the last element of the array in  $O(1)$  time. Now, perform the heapify operation which will take  $O(\log n)$  time.

$$\begin{aligned} \text{Time taken} &= O(1) + O(1) + O(\log n) \\ &= O(\log n) \end{aligned}$$

B

 $O(n)$ 

Your answer is Wrong

C

 $O(n \log n)$ 

D

 $O(n^2)$ 

QUESTION ANALYTICS

**Q. 21**



Ashima Garg

 Course: GATE  
 Computer Science Engineering(CS)

HOME

MY TEST

BOOKMARKS

MY PROFILE

REPORTS

BUY PACKAGE

ASK AN EXPERT

OFFER

 EXCLUSIVE OFFER FOR OTS  
 STUDENTS ONLY ON BOOK  
 PACKAGES

is excluded from every minimum spanning tree.  
 Which of the following is true?

Have any Doubt?

A

Only  $S_1$ 

Correct Option

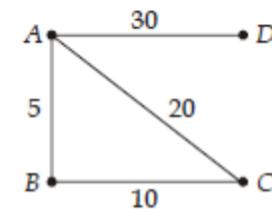
Solution :

(a)

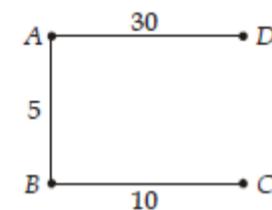
Since edges costs are distinct, so cheapest edge must be present in every minimum spanning tree while expensive edge may not be excluded from every minimum spanning tree.

Statement  $S_2$  is false

Example:



MST will be:



So, most expensive edge is not excluded.

B

Only  $S_2$ 

C

Both  $S_1$  and  $S_2$ 

Your answer is Wrong

D

Neither  $S_1$  nor  $S_2$ 

QUESTION ANALYTICS

**Q. 22**

Consider an array of numbers, where every number of array are same, is given as input. Which of the following statements is true, if we want to sort the array.

$S_1$  : Asymptotic running time of quick sort is  $O(n \log n)$ .

$S_2$  : Asymptotic running time of insertion sort is  $O(n)$ .

$S_3$  : Asymptotic running time of selection sort is  $O(n)$ .

$S_4$  : Asymptotic running time of merge sort is  $O(n \log n)$ .

Have any Doubt?

A

Only  $S_4$ 

Your answer is Wrong

B

Only  $S_2$  and  $S_4$ 

Correct Option

Solution :

(b)

$S_1$  : Fully sorted array in case of quick sort becomes worst case.

Hence, time taken =  $O(n^2)$ .

$S_2$  : Insertion sort on almost sorted array takes  $O(n)$  time.

$S_3$  : Selection sort always takes  $O(n^2)$  time.



Ashima Garg

 Course: GATE  
 Computer Science Engineering(CS)

HOME

MY TEST

BOOKMARKS

MY PROFILE

REPORTS

BUY PACKAGE

ASK AN EXPERT

OFFER

 EXCLUSIVE OFFER FOR OTS  
 STUDENTS ONLY ON BOOK  
 PACKAGES
Only  $S_1, S_2$  and  $S_4$ 

D

Only  $S_3$  and  $S_4$ 

QUESTION ANALYTICS

**Q. 23**

In an adjacency list representation of an undirected graph  $G = (v, e)$ , for any 2 sets of vertices  $v_1$  and  $v_2$  let, distance  $(v_1, v_2)$  be defined as the minimum of the length of shortest distance between a vertex in  $v_1$  and  $v_2$ , if  $v_1 \cap v_2 \neq \emptyset$ , then distance  $(v_1, v_2) = 0$ . The most optimal time complexity for computing distance  $(v_1, v_2)$  is:

A

 $O(|v| + |e|)$ Your answer is **Correct****Solution :**

(a)

Take 2 new vertices  $a$  and  $b$ . Connect  $a$  to all vertices in  $v_1$  and  $b$  to all vertices in  $v_2$ . Perform BFS from  $a$  to  $b$ . The length of the path obtained minus 2 (one edge from  $a$  and one edge to  $b$ ) will give the required result.

B

 $O(|v| \times |e|)$ 

C

 $O(|v| \times |e| + |e| \times |v|)$ 

D

 $O(|e| \times |v^2|)$ 

QUESTION ANALYTICS

**Q. 24**

Match List-I (Algorithm) with List-II (Complexity) and select the correct answer using the codes given below the lists:

**List-I (Algorithm)**

- A. Merge sort with no extra space.
- B. Finding number of inversion pair.
- C. Topological sort with ' $n$ ' edges and  $n$  vertices.
- D. Finding median of two sorted array each of size ' $n$ '.

**List-II (Complexity)**

1.  $O(n \log n)$
2.  $O(\log n)$
3.  $O(n^2)$
4.  $O(n)$

**Codes:**

	A	B	C	D
(a)	2	3	1	4
(b)	3	1	4	2
(c)	2	3	4	1
(d)	1	3	4	2

A

a

B

b



Ashima Garg

 Course: GATE  
 Computer Science Engineering(CS)

HOME

MY TEST

BOOKMARKS

MY PROFILE

REPORTS

BUY PACKAGE

ASK AN EXPERT

OFFER

 EXCLUSIVE OFFER FOR OTS  
 STUDENTS ONLY ON BOOK  
 PACKAGES

- A. Merge sort is outplace algorithm, if extra space is not allowed the it take  $O(n^2)$  time.  
 B. Finding number of inversion pair using merge sort takes  $O(n \log n)$  time.  
 C. Topological sort take  $O(n)$  time using DFS.  
 D. Finding median of two sorted array takes  $O(\log n)$  time.

C

c

D

d

## QUESTION ANALYTICS

## Q. 25

In a directed graph  $G$  (number of vertices =  $n$ ), super-sink is a vertex whose indegree is  $(n - 1)$  and outdegree is zero. What will be the optimal time complexity to determine if such a vertex exists in the graph or not if adjacency matrix representation of graph is used?

Have any Doubt?

A

 $O(n)$ 

Your answer is Correct

## Solution :

(a)

A vertex is super sink if and only if  $M[i, j] = 0 \forall j$  and  $M[j, i] = 1 \forall j \neq i$ .

Traverse over the matrix and check the above condition,

Using below program we can find super-sink in directed graph:

 $i = 0;$ 

do

{

 $j = i + 1;$ while ( $(j < n) \&& !A[i][j]$ )  $j++;$ if ( $j < n$ )  $i = j;$ } while ( $i < n$ ); $flag = 1;$ for ( $j = 0; j < n; j++$ )if ( $((j != i) \&& (A[i][j] || !A[j][i])) flag = 0;$ if ( $flag$ ) printf("Sink exists");

else printf("Sink does not exist");

B

 $O(log n)$ 

C

 $O(n^2)$ 

D

 $O(log log n)$ 

## QUESTION ANALYTICS

## Q. 26

Consider the partial pseudocode given below used to calculate median of combination of 2 sorted arrays (arr1 and arr2) of equal size n.

```
int MEDIAN(int arr1[], int arr2[], int n) {
  m1 = median(arr1) /* calculates median of array elements, If n is even, median = (middle element + next element)/2, if n is odd, median = middle element */
```



Ashima Garg

 Course: GATE  
 Computer Science Engineering(CS)

HOME

MY TEST

BOOKMARKS

MY PROFILE

REPORTS

BUY PACKAGE

ASK AN EXPERT

OFFER

 EXCLUSIVE OFFER FOR OTS  
 STUDENTS ONLY ON BOOK  
 PACKAGES

```

if (no of elements == even) return S1;
return MEDIAN(arr1 + n/2, arr2, n - n/2);
}
if (m1 < m2) {
  if (no of elements == even)
    return MEDIAN(arr2 + n/2 - 1, arr1, n - n/2 + 1);
  return S2;
}

```

The missing argument lists are respectively:

- MEDIAN(arr1 + n/2 - 1, arr2, n - n/2 + 1); MEDIAN(arr2 + n/2, arr1, n - n/2)
- MEDIAN(arr2 + n/2, arr1, n - n/2); MEDIAN(arr1 + n/2 - 1, arr2, n - n/2 + 1)
- MEDIAN(arr2 + n/2, arr1, n - n/2) ; MEDIAN(arr2 + n/2, arr1, n - n/2)
- MEDIAN(arr1 + n/2 - 1, arr2, n - n/2 + 1); MEDIAN(arr1 + n/2 - 1, arr2, n - n/2 + 1)

Have any Doubt ?

A

MEDIAN(arr1 + n/2 - 1, arr2, n - n/2 + 1); MEDIAN(arr2 + n/2, arr1, n - n/2)

Your answer is Correct

**Solution :**

(a)

If  $m_1$  is greater than  $m_2$ , then median is present in between first element of  $arr1$  to  $m_1$  or from  $m_2$  to last element of  $arr2$ . If  $m_2$  is greater than  $m_1$ , then median is present in between  $m_1$  to last element of  $arr1$  or from first element of  $arr2$  to  $m_2$ .

B

MEDIAN(arr2 + n/2, arr1, n - n/2); MEDIAN(arr1 + n/2 - 1, arr2, n - n/2 + 1)

C

MEDIAN(arr2 + n/2, arr1, n - n/2) ; MEDIAN(arr2 + n/2, arr1, n - n/2)

D

MEDIAN(arr1 + n/2 - 1, arr2, n - n/2 + 1); MEDIAN(arr1 + n/2 - 1, arr2, n - n/2 + 1)

QUESTION ANALYTICS

**Q. 27**

Consider  $G = (V, E)$  be a connected graph with  $n$  vertices and  $m$  edges with distinct positive edge weight.  $T = (V, E')$  be a spanning tree of  $G$  and bottleneck edge of  $T$  is the edge with greatest cost in  $T$ . A spanning tree  $T$  of  $G$  is a minimum bottleneck spanning tree if there is no spanning tree  $T'$  of  $G$  with cheaper bottleneck edge. Which of the following statement true?

 $S_1$ : Every minimum bottleneck tree of  $G$  is a minimum spanning tree of  $G$ . $S_2$ : Every minimum spanning tree of  $G$  is a minimum bottleneck tree of  $G$ .

Have any Doubt ?

A

Only  $S_1$ 

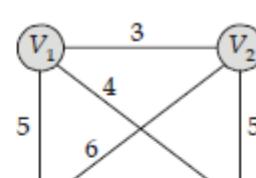
B

Only  $S_2$ 

Correct Option

**Solution :**

(b)

 $S_1$  : Consider the following graph:



Ashima Garg

 Course: GATE  
 Computer Science Engineering(CS)

HOME

MY TEST

BOOKMARKS

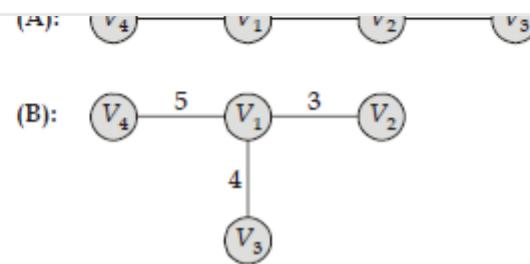
MY PROFILE

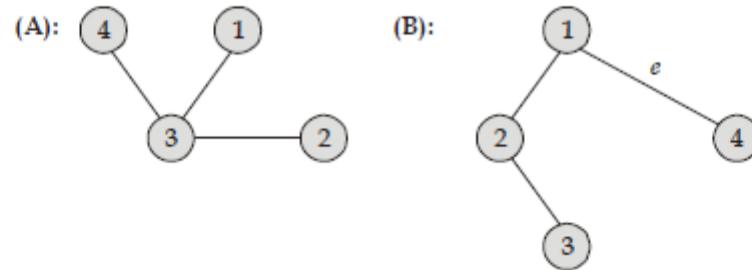
REPORTS

BUY PACKAGE

ASK AN EXPERT

OFFER

 EXCLUSIVE OFFER FOR OTS  
 STUDENTS ONLY ON BOOK  
 PACKAGES

 However only (b) is MST therefore  $S_1$  is false.

 $S_2$ : Consider 2 MST's:


Where A has a lighter bottleneck edge. This means that B has an edge 'e' that is heavier than every edge of A. If we include this edge in A, it will form a cycle in which e would be heavier. This contradicts the definition of MST.

Thus, 'e' can't be present in B, meaning that both A and B have same bottleneck edge.

Thus,  $S_2$  is true.

C

Both  $S_1$  and  $S_2$ 

D

Neither  $S_1$  nor  $S_2$ 

## QUESTION ANALYTICS

## Q. 28

Given two sequences A and B. Let  $X(A, B)$  denote the number of times that A appears as subsequence of B i.e. sequence ab appears 4 times as a subsequence of aebabdb. Let  $A_i$  denotes the first  $i$  characters of string A and  $A[i]$  denote the  $i^{\text{th}}$  character. Which of the following will computes the recurrence relation  $C(A_i, B_j)$ ?

$$C(A_i, B_j) = \begin{cases} 1; & \text{if } (i = 0) \\ 0; & \text{if } (i > 0 \text{ and } j = 0) \\ \dots & \dots \end{cases}$$

Have any Doubt?

A

 $C(A_i, B_{j-1})$ ; if  $A[i] \neq B[j]$  $C(A_i, B_{j-1}) + C(A_{i-1}, B_{j-1})$ ; if  $A[i] = B[j]$ 

Correct Option

## Solution :

(a)

$$C(A_i, B_j) = \begin{cases} 1; & \text{if } (i = 0) \\ 0; & \text{if } (i > 0 \text{ and } j = 0) \\ C(A_i, B_{j-1}); & \text{if } (A[i] \neq B[j]) \text{ when elements are not matched.} \\ C(A_i, B_{j-1}) + C(A_{i-1}, B_{j-1}); & \text{if } (A[i] = B[j]) \text{ when elements are matched.} \end{cases}$$

B

 $C(A_{i-1}, B_j)$ ; if  $A[i] = B[j]$  $C(A_i, B_{j-1}) + C(A_{i-1}, B_{j-1})$ ; if  $A[i] = B[j]$ 

C

 $C(A_i, B_j)$ ; if  $A[i] = B[j]$



Ashima Garg

 Course: GATE  
 Computer Science Engineering(CS)

HOME

MY TEST

BOOKMARKS

MY PROFILE

REPORTS

BUY PACKAGE

ASK AN EXPERT

OFFER

 EXCLUSIVE OFFER FOR OTS  
 STUDENTS ONLY ON BOOK  
 PACKAGES

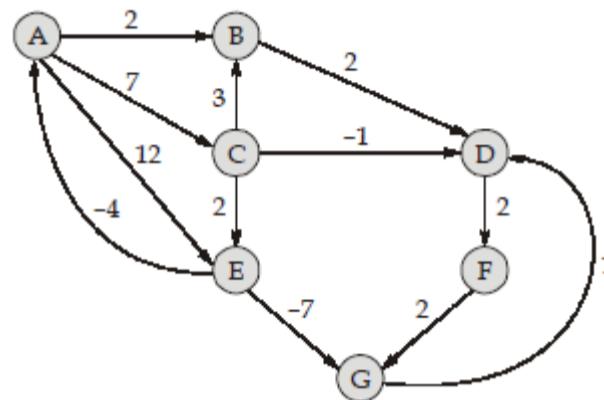
D

None of the above

## QUESTION ANALYTICS

## Q. 29

While applying Dijkstra algorithm on given graph, it is already known that it is computing wrong path to some of the vertices. The number of wrong path computed are \_\_\_\_\_.



Have any Doubt?

3

Correct Option

## Solution :

3

	Using Dijkstra algorithm	Actual Shortest path
A → B	2	AB
C	7	AC
D	4	ABD
E	9	ACE
F	6	ABDF
G	8	ABDFG

So, there are 3 wrong path calculated to D, F and G vertices.

Your Answer is 2

## QUESTION ANALYTICS

## Q. 30

Suppose to encode a text with the characters *a, b, c, d, e, f* with the following frequencies:

Letters	a	b	c	d	e	f
Frequency	15	17	25	23	18	12

If followed fixed length code each character takes 1 bits. If compression technique used is Huffman coding, then the average length of the encoded message in bits is \_\_\_\_\_. (Upto 2 decimal places)

Have any Doubt?

2.56 [2.55 - 2.57]

Correct Option

## Solution :

2.56 [2.55 - 2.57]

Total characters are 110 bits.

As Huffman tree:



Ashima Garg

Course: GATE  
Computer Science Engineering(CS)

HOME

MY TEST

BOOKMARKS

MY PROFILE

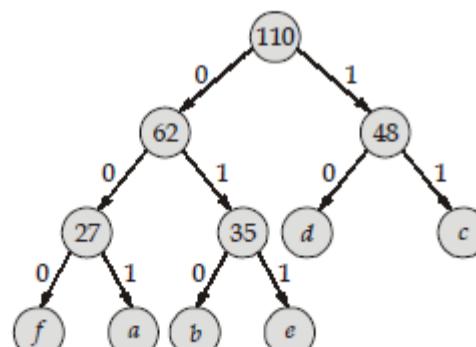
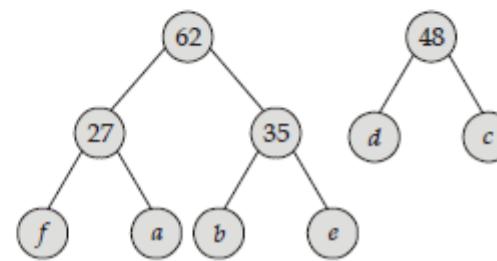
REPORTS

BUY PACKAGE

ASK AN EXPERT

OFFER

EXCLUSIVE OFFER FOR OTS STUDENTS ONLY ON BOOK PACKAGES



According to Huffman encoding

a = 001  $\Rightarrow$  3 bitb = 010  $\Rightarrow$  3 bitc = 11  $\Rightarrow$  2 bitd = 10  $\Rightarrow$  2 bite = 011  $\Rightarrow$  3 bitf = 000  $\Rightarrow$  3 bitNumber of bits needed =  $(3 \times 12) + (3 \times 15) + (3 \times 17) + (3 \times 18) + (2 \times 23) + (2 \times 25) = 282$ Average bits =  $\frac{282}{110} = 2.56$  bits/char

Your Answer is 56.4

QUESTION ANALYTICS

**Q. 31**Consider a matrix multiplication chain  $A_1, A_2, A_3$  and  $A_4$ , where matrices  $A_1, A_2, A_3$  and  $A_4$  are of dimensions  $10 \times 100, 100 \times 5, 5 \times 50$  and  $50 \times 1$  respectively.The difference between the minimum and maximum number of scalar multiplications required to find the product of  $A_1, A_2, A_3$  and  $A_4$  using basic matrix multiplication method is \_\_\_\_\_.

Have any Doubt?

73750

Your answer is Correct 73750

Solution :

73750

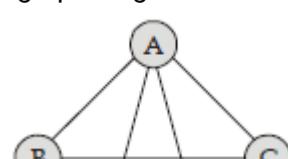
$A_1 A_2 A_3 A_4$				
$(A_1(A_2(A_3A_4)))$	$((A_1A_2)(A_3A_4))$	$((A_1A_2A_3)A_4)$	$((A_1(A_2A_3))A_4)$	$(A_1((A_2A_3)A_4))$
$A_{34} = 250$	$A_{12} = 5000$	$A_{12} = 5000$	$A_{23} = 25000$	$A_{23} = 25000$
$A_{24} = 500$	$A_{34} = 250$	$A_{13} = 2500$	$A_{13} = 50000$	$A_{24} = 5000$
$A_{14} = 1000$	$A_{14} = 50$	$A_{14} = 500$	$A_{14} = 500$	$A_{14} = 1000$
1750	5300	8000	75500	31000

$$\text{So, difference} = |\text{Maximum multiplication} - \text{Minimum multiplication}| \\ = |75500 - 1750| \\ = 73750$$

QUESTION ANALYTICS

**Q. 32**

Consider a graph 'G' given below:





Ashima Garg

 Course: GATE  
 Computer Science Engineering(CS)

HOME

MY TEST

BOOKMARKS

MY PROFILE

REPORTS

BUY PACKAGE

ASK AN EXPERT

OFFER

 EXCLUSIVE OFFER FOR OTS  
 STUDENTS ONLY ON BOOK  
 PACKAGES

If edges are having weight 1, 2, 3, 4, 5, 6, 7, 8, 9 and 10. Then maximum possible weight that a minimum weight spanning tree of  $G$  can have is \_\_\_\_\_.

Have any Doubt?

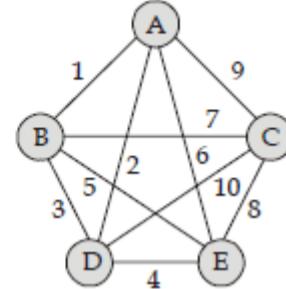
14

Correct Option

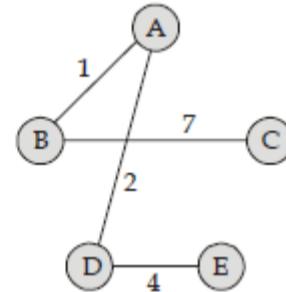
**Solution :**

14

Consider edges weights are numbered as below:



So minimum cost spanning tree will be:

In worst case, cost of MST will be  $1 + 2 + 4 + 7 = 14$ .

Your Answer is 10

QUESTION ANALYTICS

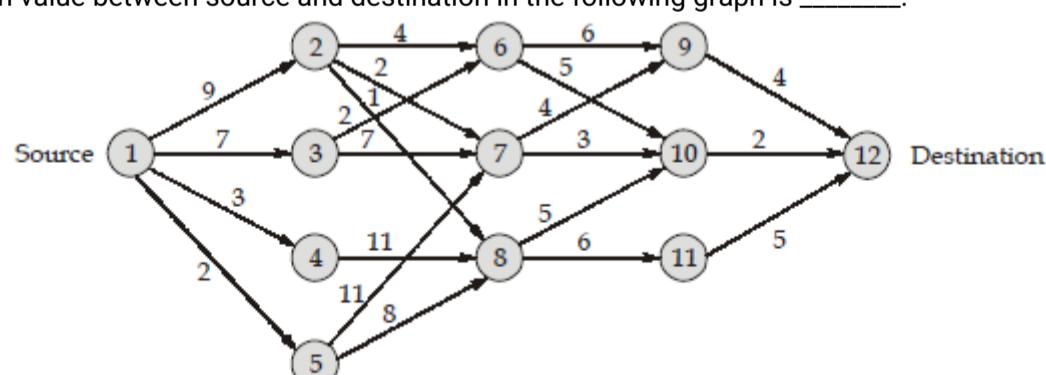
**Q. 33**

A multistage graph is a directed graph in which the nodes can be divided into a set of stages such that all edges are from a stage to next stage only. There is no edge between vertices of same stage and from a vertex of current stage to previous stage. The recurrence relation for finding the shortest path distance between two vertices  $(i, j)$  is given below:

$$\text{MSG}[i] = \min_{\forall j > i} \begin{cases} 0 & \text{if } j \in F \\ e(i, j) + \text{MSG}[j] & \text{if } j \notin F \end{cases}$$

where  $e(i, j)$  = weight of edge connecting  $i$  and  $j$ .  $F$  is the final stage.

The shortest path value between source and destination in the following graph is \_\_\_\_\_.



Have any Doubt?

16

Your answer is Correct16

**Solution :**

16

$$\text{MSG}(12) = 0 \text{ since } 12 \in F$$

$$\text{MSG}(11) = e(11, 12) + \text{MSG}(12) = 5 + 0 = 5$$

$$\text{MSG}(10) = 2$$



Ashima Garg

 Course: GATE  
 Computer Science Engineering(CS)

HOME

MY TEST

BOOKMARKS

MY PROFILE

REPORTS

BUY PACKAGE

ASK AN EXPERT

OFFER

 EXCLUSIVE OFFER FOR OTS  
 STUDENTS ONLY ON BOOK  
 PACKAGES

.....

Finally,

1	2	3	4	5	6	7	8	9	10	11	12
16	7	9	18	15	7	5	7	4	2	5	0

So, the shortest path value between source and destination is 16.

[Trick: If the graph given is multistage graph, than greedy approach can be used from backw for choosing the minimum weight, edge until reach source.]

QUESTION ANALYTICS