Kunal Jha
Course: GATE
Computer Science Engineering(CS)

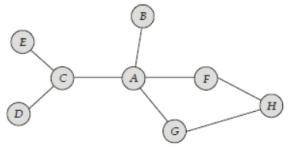
ALGORITHMS + PROGRAMMING AND DATA STRUCTURES (GATE - 2021) - REPORTS

OVERALL ANALYSIS | COMPARISON REPORT | SOLUTION REPORT

ALL(33) CORRECT(0) INCORRECT(0) SKIPPED(33)

Q. 1 FAQ | Solution Video | Have any Doubt ?

Consider the following graph. Suppose we want to traverse it, starting at node A.



Which of the following option is correct regarding these traversing orders?

I. A, G, F, B, C, D, E, H
II. A, G, H, F, C, D, E, B
III. A, C, B, F, G, H, E, D
IV. A, B, C, E, D, F, H, G

A. I and II are BFS, III and IV are DFS

B. I and III are BFS, II and IV are DFS — Correct Option

Solution :
(b)
Here are two possible orderings for BFS:
• A, G, F, B, C, D, E, H
• A, C, B, F, G, H, E, D
Here are two possible orderings for DFS:
• A, G, H, F, C, D, E, B
• A, B, C, E, D, F, H, G
So, I and III are BFS, II and IV are DFS.
Option (b) is correct.

C. I and IV are BFS, II and IV are DFS

D. II and IV are BFS, I and III are DFS

QUESTION ANALYTICS +

Q. 2 Have any Doubt ?

When a user submits a query, a search engine does the following. For every webpage that has been visited by the search engine, it computes a score indicating how relevant that page is to the query. Finally, it reports the pages with the top k scores on the screen, for a number k specified by the user. A good data structure for accumulating the scores and ranking them is:

A. A queue

B. A heap — Correct Option

Solution :
(b)
Let n be the number of pages visited by the search engine at the time a query is submitted. Assume that it takes constant time to compute the relevance score for each page w.r.t. a query. Then it takes $O(n)$ time to compute the relevance scores, a further $O(n)$ time to build a heap of n relevance scores and $O(k \cdot \log n)$ time for k delete-max operations to return the top k scores.

C. A stack

D. A binary search tree

QUESTION ANALYTICS +

Q. 3 Have any Doubt ?

Consider the program given below:

```
procedure mystery (int A[1 to 100])
int i, j, position, tmp;
begin
    for j := 1 to 100 do
        position := j;
        for i := j to 100 do
            if (A[i] > A[position]) then
            position := i;
        endfor
        tmp := A[j];
        A[j] := A[position];
        A[position] := tmp;
    endfor
end
```

When the procedure terminates, the array A has been

A Reversed

B Sorted in descending order — Correct Option

Solution :
(b)
Sorted in descending order.
At iteration j of the outer loop, the inner loop determines the index position between j and 100 such that A[position] is maximum and A[position] is interchanged with A[j]. Thus at the end, position A[j] holds the j^{th} largest value in the array.

C Left unaltered

D Sorted in ascending order

QUESTION ANALYTICS +

Q. 4 — Have any Doubt ?

Consider the program given below:

```
procedure mystery (int A[1 to 100])
int i, j, position, tmp;
begin
    for j := 1 to 100 do
        position := j;
        for i := j to 100 do
            if (A[i] > A[position]) then
            position := i;
        endfor
        tmp := A[j];
        A[j] := A[position];
        A[position] := tmp;
    endfor
end
```

Then, the number of times the test A[i] > A[position] is executed is

A 100

B 5050 — Correct Option

Solution :
(b)
In iteration j, there are 100 − j + 1 comparisons made.
So in all there are 100 + 99 + ... + 2 + 1 = 5050.

C 10000

D Depends on contents of A

QUESTION ANALYTICS +

Q. 5 — Solution Video — Have any Doubt ?

In the code fragment given below, start and end are integer values and prime(x) is a function that returns true if x is a prime number and false otherwise.
At the end of the loop what is the value of K:

```
i := 0;
j := 0;
k := 0;
for (m := start; m <= end; m := m + 1){
    k := k + m;
    if (prime(m))
    {
    i := i + m;
    } else {
    j := j + m;
    }
}
```

A $k < i + j$

B $k = i + j$ — Correct Option

Solution :
(b)
$k = i + j$
In each iteration, the value added to k is also added to exactly one of i and j.

C $k > i + j$

D Depends on start and end

QUESTION ANALYTICS +

Q. 6 Solution Video Have any Doubt ?

Consider a weighted undirected graph G with positive edge weights. Let (u, v) be an edge in the graph. It is known that the shortest path from a vertex s to u has weight 53 and the shortest path from s to v has weight 65. Which of the statement is always true?

A Weight of $(u, v) \leq 12$.

B Weight of $(u, v) = 12$.

C Weight of $(u, v) \geq 12$. Correct Option

Solution :
(c)

Weight of $(u, v) = 12$

If the weight of (u, v) is strictly less than 12, then there is a path from s to v of weight at most $53 + 11 = 64$ (which goes from s to u and then takes the edge (u, v)). This contradicts the fact that the shortest path from s to v has weight 65.
Thus the weight of (u, v) is at least 12.

D Nothing can be said about the weight of (u, v).

QUESTION ANALYTICS +

Q. 7 FAQ Have any Doubt ?

Which of the following relationships holds in general between the scope of a variable and the lifetime of a variable (in a language like C)?

A The scope of a variable is contained in the lifetime of the variable. Correct Option

Solution :
(a)
The scope of a variable is contained in the lifetime of the variable.

B The scope of a variable is the same as the lifetime of the variable.

C The lifetime of a variable is disjoint from the scope of the variable.

D None of the above

QUESTION ANALYTICS +

Q. 8 FAQ Have any Doubt ?

In the code fragment below, start and end are integer values and prime(x) is a function that returns True if x is a prime number and false otherwise.

```
i = 0;
j = 0;
k = 0;
for m = start to end {
    if prime(m) == True {
    i = i + 1;
    # Statement 1
    }
    else{
    j = j - 1;
    # Statement 2
    }
}
```

We wish to maintain the invariant $k == i - j$ after each iteration of the for loop. What should we insert at Statement 1 and Statement 2?

A Statement 1: $k = k + 1$ and Statement 2: $k = k + 1$ Correct Option

Solution :
(a)
$i+(-j)$ is the number of the times the loop is iterated, so increment k in both branches of the if.

B Statement 1: $k = k + 1$ and Statement 2: $k = k - 1$

C Statement 1: $k = k - 1$ and Statement 2: $k = k + 1$

D Statement 1: $k = k - 1$ and Statement 2: $k = k - 1$

QUESTION ANALYTICS +

Q. 9

Have any Doubt ?

How many times is the comparison ($i >= n$) performed in the following program?

```
int i = 85, n = 5;
main( ) {
    while (i >= n) {
    i = i - 1;
    n = n + 1;
    }
}
```

42 — Correct Option

Solution :
42
The value of $i - n$ is 80 initially. We run the loop as long as $i - n = 0$ and in each iteration, $i - n$ decreases by 2.
Just before the k^{th} time the comparison is performed (for $k = 1$), the value of $i - n$ is $80 - 2k + 2$.
Hence just before the forty-first comparison, the value of $i - n$ is 0.
After the forty-first comparison, the loop is executed one last time. We need to make the comparison once more to exit the loop.
Thus, the correct answer is 42.

QUESTION ANALYTICS +

Q. 10

FAQ | Solution Video | Have any Doubt ?

ScamTel has won a state government contract to connect 17 cities by high-speed fibre optic links. Each link will connect a pair of cities so that the entire network is connected —there is a path from each city to every other city. The contract requires the network to remain connected if any single link fails. What is the minimum number of links that ScamTel needs to set up?

17 — Correct Option

Solution :
17
If we connect the cities in a loop with 17 links, then even if any single link fails, we can connect the cities by traversing the loop the other way. If we only had 16 links connecting 16 cities, that would constitute a tree which would have only a single path between any pair of cities and thus get disconnected by a single link failure.
Thus, the minimum number of links that ScamTel needs to set up will be 17.

QUESTION ANALYTICS +

Kunal Jha
Course: GATE
Computer Science Engineering(CS)

ALGORITHMS + PROGRAMMING AND DATA STRUCTURES (GATE - 2021) - REPORTS

Q. 11

FAQ | Have any Doubt ?

The value printed by the following program is _____.

```
void f(int* p, int m)
{
    m = m - 5;
    *p = *p - m;
    return;
}
void main( )
{
    int i = 10, j = 5;
    f(&i, j);
    printf("%", i + j);
}
```

15 — Correct Option

Solution :
15
In f pointer p subtracting the value 0 from i so i value remains 10.
The output printed will be 10 + 5 = 15.

QUESTION ANALYTICS +

Q. 12

FAQ | Have any Doubt ?

Consider the code below, defining the function mystery:

```
mystery(a, b) {
    if (a < 0 or b < 0) return 0;
    else if (a == 0) return b + 1;
    else if (b == 0) return mystery(a - 1, 1);
    else return mystery(a - 1, mystery(a, b - 1));
}
```

What would be the value of mystery(3, 3)?

61 — Correct Option

Solution :
61
mystery(3, 0) = mystery(2, 1) = 2 + 3 = 5.
mystery(3, 1) = mystery(2, mystery(3, 0)) = mystery(2, 5) = 2 · 5 + 3 = 13.
mystery(3, 2) = mystery(2, mystery(3, 1)) = mystery(2, 13) = 2 · 13 + 3 = 29.
mystery(3, 3) = mystery(2, mystery(3, 2)) = mystery(2, 29) = 2 · 29 + 3 = 61.

QUESTION ANALYTICS +

Q. 13

FAQ | Have any Doubt ?

Consider the code below, defining the function A:

```
A(m, n, p) {
    if (p == 0) return m + n;
    else if (n == 0 && p == 1) return 0;
    else if (n == 0 && p == 2) return 1;
    else if (n == 0) return m;
    else return A(m, A(m, n - 1, p), p - 1);
}
```

What would be the value of A(2, 3, 3)?

65536 — Correct Option

Solution :
65536
A(2, 0, 3) = 2.
A(2, 1, 3) = A(2, A(2, 0, 3), 2) = A(2, 2, 2) = 2^2 = 4.
A(2, 2, 3) = A(2, A(2, 1, 3), 2) = A(2, 4, 2) = 2^4 = 16.
A(2, 3, 3) = A(2, A(2, 2, 3), 2) = A(2, 16, 2) = 2^{16} = 65536.

QUESTION ANALYTICS +

Q. 14

FAQ | Solution Video | Have any Doubt ?

Given an array arr = {45, 77, 89, 90, 94, 99, 100). What are the pivot values in the first and second level of partition in quicksort? (Pivot element position is select based on formula, pivot = $\frac{n}{2}$, where n is total element for the set of that particular array). [Assume array indexing start from 1]

A 90

B 89 Correct Option

C 94 Correct Option

D 77

YOUR ANSWER - NA | CORRECT ANSWER - b,c | STATUS - SKIPPED

Solution :

(b, c)

At 1st level:

$$\text{Pivot element} = \frac{n}{2} = \frac{7}{2} = 3$$

(45, 77, [89] (90, 94, 99, 100)

At 2nd level:

$$\text{Pivot element} = \frac{n}{2} = \frac{4}{2} = 2$$

(90) [94] (99,100)

QUESTION ANALYTICS +

Q. 15

Solution Video | Have any Doubt ?

A person wants to visit some places. He starts from a vertex and then wants to visit every vertex till it finishes from one vertex, backtracks and then explore other vertex from same vertex. What algorithm or algorithms he should use?

A Depth First Search Correct Option

B Breadth First Search

C Trim's algorithm

D Kruskal's Algorithm

YOUR ANSWER - NA | CORRECT ANSWER - a | STATUS - SKIPPED

Solution :

(a)

This is the definition of the Depth First Search. Exploring a node, then aggressively finding nodes till it is not able to find any node.

QUESTION ANALYTICS +

Q. 16

FAQ | Solution Video | Have any Doubt ?

Consider the graph M with 3 vertices. Its adjacency matrix is shown below. Which of the following is/are true?

$$M = \begin{bmatrix} 0 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \end{bmatrix}$$

A Graph M has no minimum spanning tree.

B Graph M has a unique minimum spanning trees of cost 2.

C Graph M has 3 distinct minimum spanning trees, each of cost 2. Correct Option

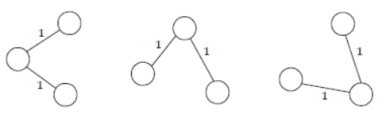
D Graph M has 3 spanning trees of different costs.

YOUR ANSWER - NA | CORRECT ANSWER - c | STATUS - SKIPPED

Solution :

(c)

Here all non-diagonal elements in the adjacency matrix are 1. So, every vertex is connected every other vertex of the graph. And, so graph M has 3 distinct minimum spanning trees.



QUESTION ANALYTICS +

Q. 17 FAQ Solution Video Have any Doubt ?

In a connected undirected graph, the distance between two vertices is the number of edges in the shortest path between them. Suppose we denote by P the following property: there exists a vertex that is a neighbour of all other vertices. Consider the following statements:
(i) If P is false, then there is a pair of vertices such that the distance between them is at least 4.
(ii) If P is true, then the distance between any pair of vertices is at most 2.
What can you say about these statements?

A Only (i) is true.

B Both (i) and (ii) are true.

C Only (ii) is true. Correct Option

Solution :
(c)
If P is true, there is a "central" vertex that is a neighbor of all others. We can start from any vertex and reach any other by going via the center. So (ii) is true. There are graphs where P is false but the distance is at most 3 between any pair of vertices, so (i) is false.

D Neither (i) nor (ii) is true.

QUESTION ANALYTICS +

Q. 18 FAQ Have any Doubt ?

Consider the code as given below. The function print( ) receives root of a Binary Search Tree (BST) and a positive integer k as arguments.

```
//A BST node
struct node {
    int data;
    struct node *left, *right;
};
int count = 0;
void print(struct node *root, int k)
    if (X)
    {
        print(Y);
        count++;
        if (count == k)
            printf("%d", root → data);
        print(root → left, k);
    }
}
```

If function print( ) prints the k^{th} largest element in BST, then what should be X and Y?

A X = root != NULL && count >= k and Y = root → right, k

B X = root != NULL && count <= k and Y = root → right

C X = root != NULL && count >= k and Y = root → right

D X = root != NULL && count <= k and Y = root → right, k Correct Option

Solution :
(d)
Option (d) is correct for X and Y places.
It correctly outputs k^{th} largest element in BST.

QUESTION ANALYTICS +

Q. 19 FAQ Solution Video Have any Doubt ?

Which of the following recurrence relation can be solved by using Master's theorem directly or indirectly?

A $T(n) = 2^n T\left(\frac{n}{2}\right) + n^n$

B $T(n) = 16T\left(\frac{n}{4}\right) + n!$ Correct Option

Solution :
(b)
(a) Does not apply (a is not constant).
(b) $T(n) = T(n!)$ (Case 3).
(c) Does not apply ($a < 1$).
(d) Does not apply (non-polynomial difference between $f(n)$ and $n^{\log_b a}$).

C $T(n) = 0.5T\left(\frac{n}{2}\right) + \frac{1}{n}$

D $T(n) = 2T\left(\frac{n}{2}\right) + \left(\frac{n}{\log n}\right)$

QUESTION ANALYTICS +

Q. 20 FAQ Solution Video Have any Doubt ?

What is the time complexity of following function fun( )? Assume that $\log(x)$ returns log value in base 2.

```
void fun( )
{
    int i, j;
    for (i = 1; i <= n; i++)
        for (j = 1; j <= log(i); j++)
            printf("IITBOMBAY");
}
```

A $T(n)$

B $T(n \log \log n)$

C $T(n \log n)$ Correct Option

Solution :
(c)

$$\begin{aligned} T.C &= \log 1 + \log 2 + \log 3 + \ldots\ldots + \log n \\ &= \log(1.2.3 \ldots n) \\ &= \log(n!) \\ &= \log(n^n) \\ &= n \log n \end{aligned}$$

Option (c) $T(n \log n)$ is correct.

D $T(n^2)$

QUESTION ANALYTICS +

Kunal Jha
Course: GATE
Computer Science Engineering(CS)

ALGORITHMS + PROGRAMMING AND DATA STRUCTURES (GATE - 2021) - REPORTS

OVERALL ANALYSIS | COMPARISON REPORT | SOLUTION REPORT

ALL(33) | CORRECT(0) | INCORRECT(0) | SKIPPED(33)

Q. 21 FAQ | Have any Doubt ?

The output of following C program is:

```
#include <stdio.h>
char str1[100];
char *fun(char str[ ])
{
    static int i = 0;
    if (*str)
    {
        fun(str + 1);
        str1[i] = *str;
        i++;
    }
    return str1;
}
int main( )
{
    char str[ ] = "GATE CS 2021 Mock Test";
    printf("%s", fun(str));
    return 0;
}
```

A. GATE CS 2021 Mock Test

B. tseT kcoM 1202 SC ETAG — Correct Option

Solution :
(b)
The function basically reverses the given string.

C. Segmentation Fault

D. tseT

QUESTION ANALYTICS +

Q. 22 Have any Doubt ?

The pre-order traversal of a binary tree is given by, 12, 8, 6, 2, 7, 9, 10, 16, 15, 19, 17, 20. Then the post-order traversal of this tree is:

A. 2, 6, 7, 8, 9, 10, 12, 15, 16, 17, 19, 20

B. 2, 7, 6, 10, 9, 8, 15, 17, 20, 19, 16, 12

C. 7, 2, 6, 8, 9, 10, 20, 17, 19, 15, 16, 12

D. Data not sufficient. — Correct Option

Solution :
(d)
Since we need also inorder traversal to construct this binary tree but inorder traversal is not provided.
Note that it is not mentioned that given tree is binary search tree, so any sequence of inorder is possible (i.e., sorted inorder is not necessary).
Option (d) is correct.

QUESTION ANALYTICS +

Q. 23 FAQ | Have any Doubt ?

A First In First Out queue is a data structure supporting the operations Enqueue, Dequeue, Print. Enqueue (X) adds the item x to the tail of the queue. Dequeue removes the element at the head of the queue and returns its value. Print prints the head of the queue. If the queue had n elements to begin with, how many operations would you need to print the queue in reverse order? Note the state of queue must remain intact after performing certain number of operations.

A. $2n - 1$

B. n

C. $n(2n - 1)$ — Correct Option

Solution :
(c)

Each time we do an X = Dequeue followed by an Enqueue (X), we move the element at the head to the tail. Repeating these two statements $n-1$ times moves the last element to the head, while preserving the order among the rest of the list. Issuing a Print instruction now prints the current head of the list (which was originally the last element). Thus we need $2(n-1)+1$ instructions to move the last element to the head and print it. At this point, the last element is what was originally the last but one element. If we now repeat the above block of $2(n-1)+1$ instructions, we print the second last element of the original queue, while bringing the last two elements of the original queue to the head. Extending this logic, we see that if repeat the block of code n times, we end up printing the queue in reverse. The number of statements required is $n(2n-1)$.

D $n(n-1)(2n-1)$

QUESTION ANALYTICS +

Q. 24 Solution Video | Have any Doubt ?

Consider the code given below, which runs insertion sort:

```
void insertionSort(int arr[], int array_size)
{
    int i, j, value;
    for (i = 1; i < array_size; i++)
    {
        value = arr[i];
        j = i;
        while (_________)
        {
            arr[j] = arr[j - 1];
            j = j - 1;
        }
        arr[j] = value;
    }
}
```

Which condition will correctly implement the while loop?

A (j > 0)||(arr[j − 1] > value)

B (j > 0) && (arr [j − 1] > value) Correct Option

Solution :
(b)
In insertion sort, the element is A[j] is inserted into the correct position in the sorted sequence A[1... j − 1].
So, condition given in (j > 0) && (arr[j − 1] > value) will implement while loop correctly.

C (j > 0) && (arr [j + 1] > value)

D (j > 0) && (arr [j + 1] < value)

QUESTION ANALYTICS +

Q. 25 FAQ | Solution Video | Have any Doubt ?

Let A be an array of n integers, sorted so that $A[1] < A[2] < \ldots < A[n]$. You are given a number x. The aim is to find out if there are indices k and l such that $A[k] + A[l] = x$. Suppose the time complexity of an algorithm is $n^a (\log n)^b$. Then what would be the value of $a + b$? _____

1 Correct Option

Solution :
1
Consider the two endpoints $A[1]$ and $A[n]$.

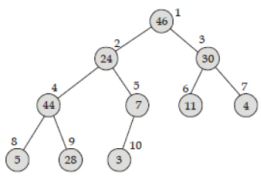
- If $A[1] + A[n] > x$, $A[n]$ is useless since it cannot be paired with any number to achieve the sum. Any sum $A[i] + A[n]$, $i > 1$, will be greater than or equal to $A[1] + A[n]$ since $A[i] \geq A[1]$.
- Likewise, if $A[1] + A[n] < x$, $A[1]$ is useless since it cannot be paired with any number to achieve the sum.

Hence, we start with two pointers left and right, with left = 1 and right = n initially. At each stage, we check the sum A[left] + A[right]. If this sum exceeds x, we decrement right and if this sum is less than x we increment left. Eventually, either left and right meet and there is no solution, or we find the indices k and l that we are looking for. This takes a single scan of A, so the overall time is $O(n)$.

QUESTION ANALYTICS +

Q. 26 Solution Video | Have any Doubt ?

The essential part of Heap sort is construction of max-heap. Consider the tree shown below, the node 24 violates the max-heap property. Once heapify procedure is applied to it, which position will it be in?



9 Correct Option

Solution :
9
In max-heap element at each node is smaller than or equal to the element at its parent node. On applying the heapify procedure on item at position 2, it will be in position 9.

QUESTION ANALYTICS +

Q. 27 Solution Video Have any Doubt ?

Consider the strings "PQRSTPQRS" and "PRATPBRQRPS". What is the length of the longest common subsequence?

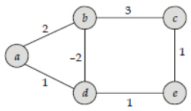
7 Correct Option

Solution :
7
The longest common subsequence is "PRTPQRS" and its length is 7.

QUESTION ANALYTICS +

Q. 28 Solution Video Have any Doubt ?

Consider the following graph:



What is the minimum cost to travel from node a to node c?

2 Correct Option

Solution :
2
The minimum cost to travel from node a to node c is 2.

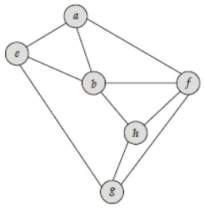
a to d, cost = 1
d to b, cost = −2
b to c, cost = 3

Hence the total cost is 2.

QUESTION ANALYTICS +

Q. 29 Solution Video Have any Doubt ?

Number of different BFS traversal starting from node b _____.



24 Correct Option

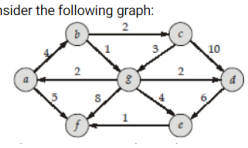
Solution :
24

$b \underbrace{[a\ e\ h\ f]}_{4!} g = 24$

QUESTION ANALYTICS +

Q. 30 Solution Video Have any Doubt ?

Consider the following graph:



If b is the source vertex, what is the minimum cost to reach f vertex? _____

6 Correct Option

Solution :
6
The minimum cost to reach f vertex from b vertex is 6 by having vertices g and e as intermediates.
b to g, cost is 1
g to e, cost is 4
e to f, cost is 1
Hence total cost 1 + 4 + 1 = 6.

QUESTION ANALYTICS +

Kunal Jha
Course: GATE
Computer Science Engineering(CS)

HOME
MY TEST
BOOKMARKS
MY PROFILE
REPORTS
BUY PACKAGE
NEWS
TEST SCHEDULE

ALGORITHMS + PROGRAMMING AND DATA STRUCTURES (GATE - 2021) - REPORTS

OVERALL ANALYSIS | COMPARISON REPORT | SOLUTION REPORT

ALL(33) | CORRECT(0) | INCORRECT(0) | SKIPPED(33)

Q. 31

FAQ | Solution Video | Have any Doubt ?

Which of the following is/are true?

A. The spanning trees do not have any cycles. — Correct Option

B. MST have n – 1 edges if the graph has n vertices. — Correct Option

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. — Correct Option

D. Removing one edge from the spanning tree will not make the graph disconnected.

YOUR ANSWER - NA | CORRECT ANSWER - a,b,c | STATUS - SKIPPED

Solution :
(a, b, c)
Every spanning tree has n – 1 edges if the graph has n vertices and has no cycles. The MST follows the cut property, 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.

QUESTION ANALYTICS +

Q. 32

Solution Video | Have any Doubt ?

Which of the following is/are true about Prim's algorithm?

A. It is a greedy algorithm. — Correct Option

B. It constructs MST by selecting edges in increasing order of their weights.

C. It never accepts cycles in the MST. — Correct Option

D. It can be implemented using the Fibonacci heap . — Correct Option

YOUR ANSWER - NA | CORRECT ANSWER - a,c,d | STATUS - SKIPPED

Solution :
(a, c, d)
Prim's algorithm can be implemented using Fibonacci heap and it never accepts cycles. Prim's algorithm follows greedy approach. Prim's algorithms span from one vertex to another.

QUESTION ANALYTICS +

Q. 33

Solution Video | Have any Doubt ?

Which of the following is/are false?

A. Prim's algorithm initializes with a vertex.

B. Prim's algorithm initializes with a edge. — Correct Option

C. Prim's algorithm initializes with a vertex which has smallest edge. — Correct Option

D. Prim's algorithm initializes with a forest. — Correct Option

YOUR ANSWER - NA | CORRECT ANSWER - b,c,d | STATUS - SKIPPED

Solution :
(b, c, d)
Steps in Prim's algorithm:
I. Select any vertex of given graph and add it to MST.
II. Add the edge of minimum weight from a vertex not in MST to the vertex in MST.
III. If MST is complete the stop, otherwise go to step (II).

QUESTION ANALYTICS +