Kunal Jha
Course: GATE
Computer Science Engineering(CS)

ALGORITHMS -1: (GATE - 2021) - REPORTS

OVERALL ANALYSIS | COMPARISON REPORT | SOLUTION REPORT

ALL(17) | CORRECT(0) | INCORRECT(0) | SKIPPED(17)

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

Insertion Sort has

A Worse case running time $\theta(n^2)$, average case running time $\theta(n^2)$ — Correct Option

Solution :
(a)
Insertion sort time complexity

$$\text{Best case} = \theta(n)$$
$$\text{Worst case} = \theta(n^2)$$
$$\text{Average case} = \frac{n+n^2}{2} = \theta(n^2)$$

B Worse case running time $\theta(n)$, average case running time $\theta(n)$

C Worse case running time $\theta(n^2)$, average case running time $\theta(n)$

D Worse case running time $\theta(n)$, average case running time $\theta(n^2)$

QUESTION ANALYTICS +

Q. 2 ? FAQ | Solution Video | Have any Doubt ?

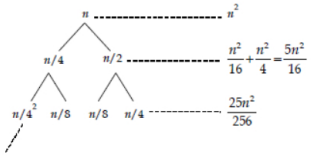
If $T(n) = T(n/4) + T(n/2) + n^2$, then the time complexity will be

A $T(n) = \theta(n)$

B $T(n) = \theta(n^2)$ — Correct Option

Solution :
(b)

$$T(n) = \underbrace{T(n/4)+T(n/2)}_{\text{Subproblems}} + \underbrace{n^2}_{\text{Cost}}$$



$$\frac{n}{4^k} = 1$$
$$n = 4^k$$
$$\log n = k \log 4$$
$$k = \frac{\log n}{\log 4} = \log 4^n$$

Hence,
$$\text{T.C} = n^2 + \left(\frac{5}{16}\right)^1 n^2 + \left(\frac{5}{16}\right)^2 n^2 \ldots\ldots \left(\frac{5}{16}\right)^{\log_4 n} n^2$$
$$\text{T.C} = n^2 \left[\underbrace{1 + \frac{5}{16} + \left(\frac{5}{16}\right)^2 + \ldots\ldots \left(\frac{5}{16}\right)^{\log_4 n}}_{\text{decreasing G.P}}\right]$$
$$\text{T.C} = \theta(n^2)$$

C $T(n) = \theta(n^3)$

D None of the above

QUESTION ANALYTICS +

Q. 3 ? FAQ | Solution Video | Have any Doubt ?

Two list of size x and y are available. The time complexity in terms of number of comparisons for the merging of the two list to make sorted list will be

A $O(x + y)$

B $O(\log x^2 + y)$

C $O(\log x + \log y)$

D $O(x \log x + y \log y)$ Correct Option

Solution :
(d)
We have to sort both the array individually before merging them we can merge in $O(x + y)$ time in worst case.
Total number of comparisons

$$= x \log x + y \log y + x + y - 1$$
$$= O(x \log x + y \log y)$$

QUESTION ANALYTICS +

Q. 4 ? FAQ ▶ Solution Video Have any Doubt ?

While sorting the following numbers using quick sort, the last number is chosen as pivot. What will be permutation of the numbers after the partition function has been applied?
70, 48, 76, 58, 43, 47, 78, 53

A 43, 48, 53, 47, 58, 70, 76, 78

B 48, 47, 43, 78, 53, 70, 76, 58

C 47, 48, 43, 76, 58, 53, 78, 70

D 48, 43, 47, 53, 70, 76, 78, 58 Correct Option

Solution :
(d)
Only option (d) is correct.

QUESTION ANALYTICS +

Q. 5 ? FAQ ▶ Solution Video Have any Doubt ?

Assume 4 sorted lists of 8 elements each one available. If we merge these list into a single sorted list. The number of key comparison in worst case using an efficient algorithm will be

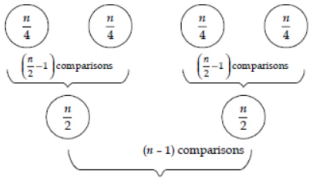
A 32

B 81

C 64

D 61 Correct Option

Solution :
(d)
Let us consider n-elements where each of the 4 list are having $\frac{n}{4}$ elements.



$$\text{Total comparison} = 2\left(\frac{n}{2} - 1\right) + n - 1 = 2n - 3$$

Here, $n = 4 \times 8 = 32$
So, Total comparison $= 2 \times 32 - 3 = 61$

QUESTION ANALYTICS +

Q. 6 ? FAQ ▶ Solution Video Have any Doubt ?

Consider the following statements in reference to sort the array of numbers.
S_1 : Asymptotic running time of selection sort is $O(n)$.
S_2 : Asymptotic running time of insertion sort is $O(n)$ is best case.
S_3 : Asymptotic running time of merge sort is $\theta(n^2)$ in worst case.

S_4 : Asymptotic running time of quick sort is O(n logn) is worst case.
The number of the correct statements is/are _____.

1 — Correct Option

Solution :
1

	Best Case	Worst Case
Selection Sort	$O(n^2)$	$O(n^2)$
Insertion Sort	$O(n)$	$O(n^2)$
Merge Sort	$O(n \log n)$	$O(n \log n)$
Quick Sort	$O(n \log n)$	$O(n^2)$

So only (1) statement S_2 is correct.

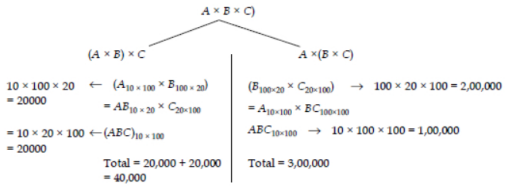
QUESTION ANALYTICS +

Q. 7 FAQ | Solution Video | Have any Doubt ?

Suppose three matrices A, B and C are given where the dimension of A is 10 × 100, B is 100 × 20 and C is 20 × 100. The minimum number of matrix multiplication required to find the product of $A \times B \times C$ using basic matrix multiplication method is _____.

40000 — Correct Option

Solution :
40000



Hence, minimum 40,000 multiplication required.

QUESTION ANALYTICS +

Q. 8 FAQ | Solution Video | Have any Doubt ?

Which of the following is/are true?

A. A binary min-heap with n elements supports INSERT in O(logn) time and DELETE-MIN in constant time.

B. The quick sort algorithm that uses linear-time median finding algorithm takes $\theta(n^2)$ time in worst-case.

C. The solution to the recurrence T(n) = $T(n) = T\left(\frac{n}{6}\right) + T\left(\frac{7n}{9}\right) + O(n)$ is $O(n)$. (Assume $T(n) = 1$ for n smaller than some constant c). — Correct Option

D. If $f(n) = \theta(g(n))$ and $g(n) = \theta(h(n))$, then $h(n) = \theta(f(n))$. — Correct Option

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

Solution :
(c, d)
False
False. It can be implemented with $O(\log n)$ auxiliary space.
True. Using the substitution method:

$$T(n) \le \frac{cn}{6} + \frac{7cn}{9} + an$$
$$\le \frac{17cn}{18} + an$$
$$\le cn - \left(\frac{cn}{18} - an\right)$$

This holds if $\frac{c}{18} - a \ge 0$, so it holds for any constant c such that $c \ge 18a$.

It can also be solve by using a recursion tree, noting that the total work at level i is $\left(\frac{17}{18}\right)^{i*n}$, which converges to $O(n)$.
True. Θ is transitive.

QUESTION ANALYTICS +

Q. 9

Which of the following is/are true?

A. If $f(n) = O(g(n))$ and $g(n) = O(h(n))$, then $h(n) = \Omega(f(n))$ — Correct Option

B. If $f(n) = O(g(n))$ and $g(n) = O(f(n))$ then $f(n) = \theta(g(n))$ — Correct Option

C. The sequence < 20, 15, 18, 7, 9, 5, 12, 3, 6, 2 > is a max-heap. — Correct Option

D. Suppose that instead of using Build-Heap to build a max-heap in place, the Insert operation is used n times. Starting with an empty heap, for each element, use Insert to insert it into the heap. After each insertion, the heap still has the max-heap property, so after n Insert operations, it is a max-heap on the n elements. Then this heap construction runs in O(n logn) time. — Correct Option

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

Solution :

(a, b, c, d)

True. O is transitive, and $h(n) = \Omega(f(n))$ is the same as $f(n) = O(h(n))$.

True.

True. For every node with 1-based index $i > 1$, the node with index floor $(i/2)$ is larger.

True. Insert takes $O(\log n)$ time per operation, and gets called $O(n)$ times.

QUESTION ANALYTICS +

Q. 10

? FAQ | Solution Video | Have any Doubt ?

Which of the following is correct?

A. $n^{\sqrt{n}} < \frac{e^{n\log n}}{n} < 2^{n\log n}$ — Correct Option

Solution :

(a)

- $\frac{e^{n\log n}}{n} = \frac{e^{\log_e n^n}}{n} = \frac{n^n}{n} = n^{n-1}$
- $2^{n\log n} = 2^{\log_2 n^n} = n^n$

Clearly, $n^{\sqrt{n}} < n^{n-1} < n^n$

Hence, option (a) is correct.

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

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

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

QUESTION ANALYTICS +

Kunal Jha
Course: GATE
Computer Science Engineering(CS)

ALGORITHMS -1: (GATE - 2021) - REPORTS

OVERALL ANALYSIS | COMPARISON REPORT | SOLUTION REPORT

ALL(17) | CORRECT(0) | INCORRECT(0) | SKIPPED(17)

Q. 11 Solution Video | Have any Doubt ?

Consider the following pseudo C program:

```
void gate (int m)
{
    int i, J, Count = 0;
    for (i = 1; i ≤ m; i ++)
{
    for (J = 1; J × J ≤ m; J++)
    Count+ = 10;
}
}
```

The complexity of the above program is

A $0(m^{3/2})$ — Correct Option

Solution :
(a)

$$\text{for } (i = 1; i \le m; i + +) \Rightarrow 0(m)$$

$$\text{for } (J = 1; J \times J \le m; J + +) \Rightarrow 0(\sqrt{m})$$

$$\text{T.C. } = 0(m \times m^{1/2}) = 0\ (m^{3/2})$$

B $0(m^2)$

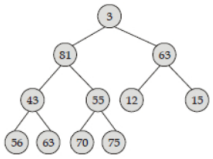
C $0(m \log m)$

D $0(m \log \log m)$

QUESTION ANALYTICS +

Q. 12 FAQ | Solution Video | Have any Doubt ?

Assume a scenario in which min-max heap has the following property: "An 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.



The worst case time complexity of deleting minimum and maximum element will be:

A $O(n)$ and $O(n)$ respectively

B O(log n) and O(log n) respectively — Correct Option

Solution :
(b)
delete max ( ) :
The max element is one out of the two children of root. Find it and the follow the same steps as delete min( ) operation. This operation will take log n time.
delete min ( ) :
Root element is the smallest element. Remove it and swap it with the right most 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 sub tree rooted at index m to maintain the min-max property.

C O(n log n) and O(n log n) respectively

D O(n) for max and O(log n) for min or O(n) for min and O(log n) for max respectively

QUESTION ANALYTICS +

Q. 13 FAQ | Solution Video | Have any Doubt ?

What is time complexity of $f( )$?

```
int f(int n)
{
    Int count = 0;
    for (int i = n; i > 0; i/= 2)
    for (int J = 1; J < i; J+= 2)
        count = 10;
    return count;
```

```
}
```

A $O(n)$ — Correct Option

Solution :
(a)

For n, inner loop execute for $\frac{n}{2}$ times.

For $\frac{n}{2}$, inner loop execute for $\frac{n}{4}$ times.

For $\frac{n}{4}$, inner loop execute for $\frac{n}{8}$ times.

$$\text{So,} \quad \text{T.C. } (n) = \frac{n}{2}+\frac{n}{4}+\frac{n}{8}+.....+1$$

$$= n\left(\frac{1}{2}+\frac{1}{4}+.....+1\right)$$

$$= O(n)$$

B $O(n \log n)$

C $O(n^2)$

D $O(n(\log n)^2)$

QUESTION ANALYTICS +

Q. 14 ? FAQ ▶ Solution Video Have any Doubt ?

Consider the following statements:
S_1 : Ternary search is more efficient than binary search for finding an element in a sorted array.
S_2 : Worst case deletion and insertion in an AVL tree has same complexity.
S_3 : The value of $T(n) = T(\sqrt{n}) + \log n$ for $n \geq 2$ is $\theta(\log n)$
The number of correct statement(s) is/are _____.

2 — Correct Option

Solution :
2

- 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 comparison in binary search : $2 \log_2 n + 1$
  Number of comparisons in ternary search = $4 \log_3 n + 1$
  Since $(2 \log_2 n + 1) < (4 \log_3 n + 1)$
  Hence binary search is preferred. So, S_1 is incorrect.
- In AVL tree, worst case deletion and insertion. Both have $O(\log n)$ complexity.
- Statement S_3 is also correct.

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

$$n = 2^m$$

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

Put $T(2^m) = S(m)$

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

By applying Master theorem, we get

$$S(m) = \theta(m)$$

$$m = \log_2 n \ (2^m = n,\ m = \log_2 n)$$

$$\text{So,} \quad T(n) = \theta(\log n)$$

QUESTION ANALYTICS +

Q. 15 ? FAQ ▶ Solution Video Have any Doubt ?

Consider the following statements:
I. $f(n) = O(g(n))$ implies $g(n) = O(f(n))$
II. $f(n) = \theta(g(n))$ implies $g(n) = \theta(f(n))$
III. If $f(n) = O(g(n))$ and $g(n) = O(h(n))$ then $f(n) = O(h(n))$
The number of statements holds true are _____.

2 — Correct Option

Solution :
2
I is false, II and III are true because I is not symmetric.

QUESTION ANALYTICS +

? FAQ ▶ Solution Video Have any Doubt ?

Q. 16

Which of the following is/are true?

A. Binary insertion sorting (insertion sort that uses binary search to find each insertion point) requires O(n logn) total operations.

B. In the merge-sort execution tree, roughly the same amount of work is done at each level of the tree. Correct Option

C. In a BST, we can find the next smallest element to a given element in O(1) time.

D. In an AVL tree, during the insert operation there are at most two rotations needed. Correct Option

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

Solution :

(b, d)

False. While binary insertion sorting improves the time it takes to find the right position for the next element being inserted, it may still take $O(n)$ time to perform the swaps necessary to shift it into place. This results in an $O(n^2)$ running time, the same as that of insertion sort.
True. At the top level, roughly n work is done to merge all n elements.

At the next level, there are two branches, each doing roughly $\frac{n}{2}$ work to merge $\frac{n}{2}$ elements. In total, roughly n work is done on that level. This pattern continues on through to the leaves, where a constant amount of work is done on n leaves, resulting in roughly n work being done on the leaf level, as well.
False. Finding the next smallest element, the predecessor, may require traveling down the height of the tree, making the running time $O(h)$.
True. The AVL property is restored on every operation. Therefore, inserting another item will require at most two rotations to restore the balance.

QUESTION ANALYTICS +

Q. 17 ? FAQ | Solution Video | Have any Doubt ?

Which of the following is/are true?

A. Counting sort is a stable, in-place sorting algorithm.

B. Given a connected graph G = (V, E), if a vertex $v \in V$ is visited during level k of a breadth-first search from source vertex $s \in V$, then every path from s to v in graph has length at most k.

C. You are given functions f and g such that f(n) = O(g(n)). Is $f(n) * \log_2(f(n)c) = O(g(n) * \log_2(g(n)))$? (Here c is some positive constant.) You should assume that f and g are non decreasing and always bigger than 1. Correct Option

D. Assume again two (positive) non decreasing functions f and g such that f(n) = O(g(n)). Is $2^{f(n)} = O(2^{g(n)})$?

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

Solution :

(c)

False. Counting sort is stable. It is not in-place, however, since we must make additional space to store the counts of the various elements. This space requirement grows as the size of the input increases. Additionally, we have to make a separate output array to produce the answer using counting sort.
False. The level of a vertex only provides the length of the shortest path from s.
True.
False:
Case 1: $f(n) = g(n) = n$. In this case, it $2^n = O(2^n)$ is certainly true.
Case 2: $f(n)=10n$, $g(n) = n$. In this case, we have $2^{(10n)} \neq O(2^n)$.

QUESTION ANALYTICS +

Kunal Jha
Course: GATE
Computer Science Engineering(CS)

ALGORITHMS-2: (GATE - 2021) - REPORTS

OVERALL ANALYSIS | COMPARISON REPORT | SOLUTION REPORT

ALL(17) | CORRECT(0) | INCORRECT(0) | SKIPPED(17)

Q. 1

FAQ | Solution Video | Have any Doubt ?

Two List-I and List-II are given. List-I basically heads the dynamic algorithm and List-II running time of the algorithm. Select the best matching options from the list.

List-I	List-II
A. 0/1 Knapsack	1. $0(mn)$
B. Matrix chain multiplication	2. $0(n)$
C. Fibonacci Series	3. $0(n^3)$
D. Travelling salesman problem	4. $0(n^n)$

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

A a

B b — Correct Option

Solution :
(b)

- 0/1 Knapsack – $0(mn)$
- Matrix chain multiplication – $0(n^3)$
- Fibonacci series – $0(n)$
- Travelling salesman – $0(n^n)$

C c

D d

QUESTION ANALYTICS +

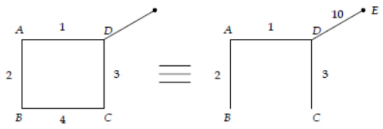
Q. 2

FAQ | Solution Video | Have any Doubt ?

Consider the below statements:
S_1 : In a connected undirected graph $G = (V, E)$ with distinct edge costs, then minimum cost edge belongs to every minimum spanning tree.
S_2 : In a connected undirected graph $G = (V, E)$ with distinct edge costs, the maximum cost edge is excluded from every minimum spanning tree.
Which of the above is correct?

A Only S_1 — Correct Option

Solution :
(a)
If the weights are unique, then every minimum spanning tree V contains minimum edge weight.
But statement S_2 is false.



Clearly, maximum edge weight is in MST.

B Only S_2

C Both S1 and S2

D Neither S_1 nor S_2

QUESTION ANALYTICS +

Q. 3

Solution Video | Have any Doubt ?

For a hash table T with n slots and m elements, load factor for T is defined by:

A n/m

B m/n — Correct Option

Solution :
(b)

C $1/mn$

D nm

QUESTION ANALYTICS +

Q. 4 FAQ | Solution Video | Have any Doubt ?

Which of the following is more suitable to find the longest path from a given vertex to any other given vertex in a directed acyclic graph (weighted) with few negative edge weights?

A Greedy Approach

B Divide and Conquer

C Hit and Trial Method

D Dynamic Programming — Correct Option

Solution :
(d)
Shortest path or longest path computation is possible using dynamic programming for directed graphs with presence of negative edge weights.

QUESTION ANALYTICS +

Q. 5 FAQ | Solution Video | Have any Doubt ?

In a graph with n vertices having degree sequence $d_1, d_2, d_3, \ldots, d_n$. What is the worst case time complexity to determine if a simple graph can be constructed or not using some algorithm?

A $0(\log n)$

B $0(n)$

C $0(n \log n)$

D $0(n^2 \log n)$ — Correct Option

Solution :
(d)
• Havel-Hakimi algorithm is used when degree sequence is given.
There are n vertices. For every vertices we will have to sort the degree sequence for each vertices, i.e., n times.
So to sort best algorithm takes $0(n \log n)$ times.
So, for n times it will be $0(n^2 \log n)$.

QUESTION ANALYTICS +

Q. 6 FAQ | Solution Video | Have any Doubt ?

A hash table is given below. The left side of figure shows the index number and right side content is the actual data stored using some hash function with linear probing.

0	m
1	n
2	
3	f
4	g
5	h
6	
7	k
8	a

The number of comparisons required in the worst case when the data is not present in the table is _____.

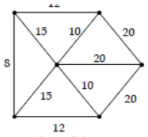
5 — Correct Option

Solution :
5
In worst case, if comparisons when < k, a, m, n, empty slot > is checked. So, total 5 comparisons are required.

QUESTION ANALYTICS +

Q. 7 FAQ | Solution Video | Have any Doubt ?

Consider the following graph:



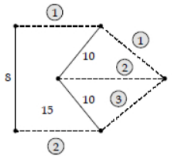
The number of distinct minimum spanning trees possible for the above is/are _____.

6 — Correct Option

Solution :

6

All minimum cost edge will be present in MST if not involve in a cycle.



So, total possible ways of selecting the edges to form spanning tree = 2 × 3 = 6.

QUESTION ANALYTICS +

Q. 8 ? FAQ ▶ Solution Video Have any Doubt ?

Which of the following statement(s) is/are true?

A. Negating all the edge weights in a weighted undirected graph G and then finding the minimum spanning tree gives us the maximum-weight spanning tree of the original graph G. — Correct Option

B. In a graph with unique edge weights, the spanning tree of second lowest weight is unique.

C. Suppose algorithm A has two steps, and A succeeds if both the steps succeed. If the two steps succeed with probability p_1 and p_2 respectively, then A succeeds with probability $p_1 p_2$. — Correct Option

D. In a connected, weighted graph, every lowest weight edge is always in some minimum spanning tree. — Correct Option

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

Solution :

(a, c,d)
(a) True.
(b) False, can construct counter-example.
(c) True.
(d) True. It can be the first edge added by Kruskal's algorithm.

QUESTION ANALYTICS +

Q. 9 ? FAQ ▶ Solution Video Have any Doubt ?

Which of the following statement(s) is/are true?

A. For a connected, weighted graph with n vertices and exactly n edges, it is possible to find a minimum spanning tree in $O(n)$ time. — Correct Option

B. In a simple, undirected, connected, weighted graph with at least three vertices and unique edge weights, the heaviest edge in the graph is in no minimum spanning tree.

C. Given a hash table with more slots than keys, and collision resolution by chaining, the worst case running time of a lookup is constant time.

D. If the DFS finishing time $f[u] > f[v]$ for two vertices u and v in a directed graph G, and u and v are in the same DFS tree in the DFS forest, then u is an ancestor of v in the depth first tree.

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

Solution :

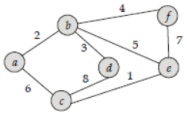
(a)

True. This graph only contains one cycle, which can be found by a DFS. Just remove the heaviest edge in that cycle.
False. If the heaviest edge in the graph is the only edge connecting some vertex to the rest of the graph, then it must be in every minimum spanning tree.
False. In the worst case we get unlucky and all the keys hash to the same slot, for $\Theta(n)$ time.
False. In a graph with three nodes, r u and v, with edges (r, u) and (r, v), and r is the starting point for the DFS, u and v are siblings in the DFS tree, neither as the ancestor of the other.

QUESTION ANALYTICS +

Q. 10 ? FAQ ▶ Solution Video Have any Doubt ?

Consider the following graph:



Which one of the following represents the sequence of edges added in order to make a minimum spanning tree using Prim's algorithm?

A (e, c), (f, c), (b, f), (b, d), (b, a)

B (a, b), (b, d), (b, e), (e, c), (b, f)

C (c, e), (e, f), (f, b), (b, d), (a, c)

D (c, e), (e, b), (b, a), (b, d), (b, f) Correct Option

Solution :
(d)
Prim's algorithm always gives connected whenever a spanning tree is constructed.
(c, e), (e, b), (b, a), (b, d), (b, f)

QUESTION ANALYTICS +

Kunal Jha
Course: GATE
Computer Science Engineering(CS)

ALGORITHMS-2: (GATE - 2021) - REPORTS

Q. 11 ? FAQ | Solution Video | Have any Doubt ?

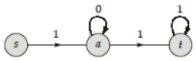
Which of the following is true?

A. Changing the Relax function to update $d[V] \geq d[u] + w(u, V)$ (instead of strictly greater than) may produce shortest path, but will not affect the correctness of Bellman-Ford algorithms outputs.

B. The running time of Radix sort is effectively independent of whether the input is already sorted or not. — Correct Option

Solution :
(b)

- The parent pointers may not lead back to the source node if a zero length cycle exists. Take an example [π means parent]



Relaxing the (S, a) edge will set $d[a] = 1$ and $\pi[a] = S$. Then relaxing the (a, a) edge will set $d[a] = 1$ and $\pi[a] = a$.
Following the π-pointers from t will no longer give a depth to S. So, it is no correct algorithm.

- Option (b) correctly states about Radix sort.

C. Both (a) and (b)

D. None of these

QUESTION ANALYTICS +

Q. 12 ? FAQ | Solution Video | Have any Doubt ?

The Subset-Sum problem is defined as follows :
Given a set of n positive integers S = {a_1, a_2, a_3, ..., a_n} and positive integer N and asked to find is there a subset whose sum is N? A dynamic program for solving this problem uses a 2-dimensional Boolean array X with n rows and $N+1$ columns. $X[i, J]; 1 \leq i \leq n; 0 \leq J \leq N$ is true if and only if there is a subset of $\{a_1, a_2, a_3, ... a_i\}$ whose elements sum to J.
Which of the following is valid for $2 \leq i \leq n$ and $a_i \leq J \leq N$?

A. $X[i, J] = X[i - 1, J]$ OR $X[i - 1, J - a_i]$ — Correct Option

Solution :
(a)

B. $X[i, J] = X[i - 1, J]$ AND $X[i, J - a_i]$

C. $X[i, J] = X[i - 1, J]$ OR $X[i, J - a_i]$

D. $X[i, J] = X[i - 1, J]$ AND $X[i - 1, J - a_i]$

QUESTION ANALYTICS +

Q. 13 ? FAQ | Solution Video | Have any Doubt ?

We are given a simple undirected graph G which has m nodes and n edges, such that $m - n = 1$. The time complexity of the most efficient algorithm in the most efficient algorithm in the worstcase to determine whether G is connected or not is equal to

A. $\theta(m + n)$ — Correct Option

Solution :
(a)
The condition can be simplified as, $n = m - 1$, i.e., the number of edges = number of vertices − 1. But this does not mean that G is a tree because even a disconnected graph can satisfy the given condition. So, option (a) is the best possible answer as we will have to use BFS.

B. $\theta(mn)$

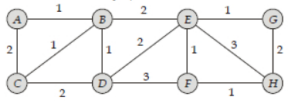
C. $\theta(m \log n + n \log m)$

D. None of these

QUESTION ANALYTICS +

Q. 14 ? FAQ ▶ Solution Video Have any Doubt ?

Consider the below graph:



Assume that E_{SP} be the cost of edges of shortest path from A to F, which is computed by Dijkstra's Algorithm and Emax is the summation of cost of all edges which show shortest path from A to every other vertices. The value of $(E_{max} - E_{SP})$ is _____.

17 Correct Option

Solution :
17
Using Dijkstra's

$$E_{SP} = (AB, BE, EF)$$
$$= 1 + 2 + 1 = 4$$
$$E_{max} = 0 + 1 + 2 + 2 + 3 + 4 + 4 + 5 = 21$$
$$E_{max} - E_{SP} = 21 - 4 = 17$$

QUESTION ANALYTICS +

Q. 15 ? FAQ ▶ Solution Video Have any Doubt ?

Consider the below statements:
S_1 : Bellman Ford algorithm for every weighted graph which contain two vertices u and v always produce a shortest path.
S_2 : If a graph contain a negative weight cycle then Dijkstra's algorithm may or may not terminate.
The number of correct statements are _____.

0 Correct Option

Solution :
0
Both statements are false.
S_1: Bellman Ford algorithm produce a shortest path if the vertices are reachable.
S_2: Dijkstra's Algorithm always terminate irrespective of the negative weight cycle. But may /may not produce correct results.

QUESTION ANALYTICS +

Q. 16 ? FAQ ▶ Solution Video Have any Doubt ?

Which of the following statement(s) is/are true?

A Let P be a shortest path from some vertex s to some other vertex t in a graph. If the weight of each edge in the graph is increased by one, P will still be a shortest path from s to t.

B Dijkstra's algorithm works on any graph without negative weight cycles.

C The Relax function never increases any shortest path estimate $d[v]$. Correct Option

D Any Dynamic Programming algorithm with n subproblems will run in O(n) time.

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

Solution :
(c)
False. In a graph where $w(s, v\,1) = 1$, $w(v\,1, v\,2) = 1$, $w(v\,2, t) = 1$ and $w(s, t) = 4$, the shortest path would change if 1 was added to every edge weight.
False. A single negative edge makes false the assumption that when you expand a node, you have already found the shortest path to that node.
True. It only changes $d[v]$ if it can decrease it.
False. The subproblems may take longer than constant time to compute, as was the case with longest increasing subsequence.

QUESTION ANALYTICS +

Q. 17 ? FAQ ▶ Solution Video Have any Doubt ?

Which of the following statement(s) is/are false?

A In a weighted undirected graph $G = (V, E, w)$, breadth-first search from a vertex s finds single-source shortest paths from s (via parent pointers) in O(V + E) time. Correct Option

B If a graph represents tasks and their interdependencies (i.e., an edge (u, v) indicates that u must happen before v happens), then the breadth-first search order of vertices is a valid order in which to tackle the tasks. Correct Option

C Dijkstra's shortest-path algorithm may relax an edge more than once in a graph with a cycle. Correct Option

D Given a weighted directed graph $G = (V, E, w)$ and a source $s \in V$, if G has a negative-weight cycle somewhere, then the Bellman-Ford algorithm will necessarily compute an incorrect result for some $\delta(s, v)$. Correct Option

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

Solution :

(a, b, c, d)
False. Only in unweighted graphs.
False, you'd prefer depth-first search, which can easily be used to produce a topological sort of the graph, which would correspond to a valid task order. BFS can produce incorrect results. False. Dijkstra's algorithm always visits each node at most once; this is why it produces an incorrect result in the presence of negative-weight edges. False. The negative-weight cycle has to be reachable from s.

QUESTION ANALYTICS +

Kunal Jha
Course: GATE
Computer Science Engineering(CS)

ALGORITHMS (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 statements:
I. $n^{3600} = O(n^{3601})$
II. $n^{3602} = O(n^{3601})$
Which of the above statement(s) is/are correct?

A Only I — Correct Option

Solution :
(a)
- I is easy to see and it is correct.
- II is incorrect. Let's see how.
  - $O(n^{3601})$ contains all functions asymptotically smaller than n^{3601}.

But, $n^{3602} \notin O(n^{3600})$
So, statement II is incorrect.

B Only II

C Both I and II

D None of these

QUESTION ANALYTICS +

Q. 2 FAQ | Solution Video | Have any Doubt ?

An array *A* in which every element is either 5 or 6. The time complexity of the most efficient algorithm which sorts A in descending order is equal to

A $O(\log n)$

B $O(n)$ — Correct Option

Solution :
(b)
• The simplest way is to scan the entire array once and maintain a count variable to count number of 5's and 6's.
• After counting number of 5 and 6's fill the array with 5 and 6 based on number of 5's and 6 with 5 being filled first then 6 will be filled.

C $O(n \log n)$

D $O(n^2)$

QUESTION ANALYTICS +

Q. 3 FAQ | Solution Video | Have any Doubt ?

Which of the following is incorrect about merge sort?

A Merge sort time complexity is same for both array and linked list.

B Merge procedure takes $O(m+n)$ time to merge two sorted arrays of size m and n.

C Merge sort on linked test is an in-place algorithm.

D None of these — Correct Option

Solution :
(d)
• Option (a) is correct, for both time complexity is $O(n \log n)$.
• Option (b) and (c) are also correct.

QUESTION ANALYTICS +

Q. 4 FAQ | Solution Video | Have any Doubt ?

Consider you are placing game of shooting balloon and you are expected to shoot 'n' balloons 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?

A O(1)

B O(n) Correct Option

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. 5 ? FAQ ▶ Solution Video Have any Doubt ?

Consider the following statement:
S_1 : Ternary search is more efficient than binary search for finding an element in a sorted array.
S_2 : Worst case deletion and insertion in an AVL tree has same complexity.

A S_1 is correct, S_2 is not.

B S_2 is correct, S_1 is not. Correct Option

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. 6 ? FAQ ▶ Solution Video Have any Doubt ?

A hash table of size 10, is shown in the figure with symbols stored from a to g using some hash function with linear probing. The worst case number of probes required when the symbol being searched is not in the table is?

0	a
1	b
2	c
3	
4	d
5	
6	e
7	f
8	g
9	

A 5

B 4 Correct Option

Solution :
(b)
The search using linear probing stops whenever it finds an empty slot.
$\therefore$ (e, f, g and then empty slot) and (a, b, c and the empty slot) are worst case searches. Both searches above has 4 comparisons.

C 3

D 7

QUESTION ANALYTICS

Q. 7 Solution Video | Have any Doubt ?

Which one of the following correctly determines the solution of the recurrence relation with $T(1) = 1$?

$$T(n) = 2T\left(\frac{n}{4}\right) + \sqrt{n}$$

A $O(n)$

B $O(\sqrt{n}\log n)$ — Correct Option

Solution :
(b)
Apply Master Theorem :

$$T(n) = aT\left(\frac{n}{b}\right) + f(n)$$

$$f(n) = n^{1/2}$$

and here $a = 2, b = 4$

So, $(n^{\log_b a}) = (n^{\log_4 2})$

Will gives $(n^{1/2})$

So, $f(n) = \theta(n^{1/2})$

So, $T(n) = O(\sqrt{n}\log n)$

C $O(n^2)$

D $O(\log n)$

QUESTION ANALYTICS

Q. 8 FAQ | Solution Video | Have any Doubt ?

Let $A(n)$ denotes the number of n bit binary strings which have no pair of consecutive 1's. Then the time complexity of the efficient algorithm which uses dynamic programming to compute $A(n)$ will be

A $O(n)$ — Correct Option

Solution :
(a)
If we see the pattern carefully by taking small values, it always follows that $A(n)$ is always the sum of $A(n-1)$ and $A(n-2)$.
Hence, $A(n)$ is equivalent to the n^{th} Fibonacci number.
Thus, using dynamic programming, the time complexity will be $O(n)$.

B $O(2^n)$

C $O(n^3)$

D $O(n \log n)$

QUESTION ANALYTICS

Q. 9 Solution Video | Have any Doubt ?

Consider the following matrices with dimensions:
A_1 is 4×6
A_2 is 6×8
A_3 is 8×4
A_4 is 4×5
The optimal number of multiplication required for $(A_1 \times A_2 \times A_3 \times A_4)$ is _____.

368 — Correct Option

Solution :
368
(a) $((A_1A_2)A_3)A_4$ requires $(4 \times 6 \times 8 + 4 \times 8 \times 4 + 4 \times 4 \times 5) = 400$ multiplications.
(b) $(A_1(A_2A_3))A_4$ requires $(6 \times 8 \times 4 + 4 \times 6 \times 4 + 4 \times 4 \times 5) = 368$ multiplications.
(c) $A_1((A_2A_3)A_4$ requires $(6 \times 8 \times 4 + 6 \times 4 \times 5 + 4 \times 6 \times 5) = 432$ multiplications.
(d) $(A_1A_2)(A_3A_4)$ requires $(4 \times 6 \times 8 + 8 \times 4 \times 5 + 4 \times 8 \times 5) = 512$ multiplications.

QUESTION ANALYTICS

Q. 10

? FAQ | Solution Video | Have any Doubt ?

Consider the Knapsack instance below:

Capacity of Knapsack = 15

Number of objects = 6, i.e., $(O_1, O_2, ..., O_6)$

Profit $(P_1, P_2, ... P_6)$ = (8, 5, 3, 15, 6, 18)

Weight (w_1, w_2, w_6) = (3, 4, 6, 4, 5, 2)

Knapsack problem is solved using maximum, profit per unit weight. How much partial weight is taken from any object? _____

2 — Correct Option

Solution :

2

$\frac{P_i}{w_i}$ ratio $\Rightarrow (O_1, O_2, ..., O_6)$ = (2.6, 1.25, 0.5, 3.75, 1.2, 9)

$\frac{P_i}{w_i}$ (in decreasing order) = (9, 3.75, 2.6, 1.25, 1.2, 0.9)

$= (O_6, O_4, O_1, O_2, O_5, O_3)$

If we go on including the objects in the above order, then O_5 is partially placed in the Knapsack and only 2 unit of the object O_5 is placed partially.

QUESTION ANALYTICS +

Kunal Jha
Course: GATE
Computer Science Engineering(CS)

ALGORITHMS (GATE - 2021) - REPORTS

OVERALL ANALYSIS | COMPARISON REPORT | SOLUTION REPORT

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

Q. 11 Solution Video | Have any Doubt ?

Given are 2 sequences *X* and *Y* respectively as *X* = "*ABCDGH*" and *Y* = "*AEDFHR*". The length of the longest common subsequence present in both X and Y will be _____.

3 — Correct Option

Solution :
3
Longest common subsequence = "*ADH*".
Hence, length of LCS = 3.

QUESTION ANALYTICS +

Q. 12 FAQ | Solution Video | Have any Doubt ?

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 _____.

3 — Correct Option

Solution :
3
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 +

Q. 13 FAQ | Solution Video | Have any Doubt ?

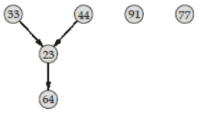
Consider an initially empty hash table of length 10. Following set of keys are inserted using open addressing with hash function $h(k) = k \bmod 10$ and linear probing.

	0
91	1
	2
33	3
44	4
23	5
64	6
77	7
	8
	9

The number of different insertion sequence of the key values using the given hash function and linear probing will result in the hash table shown in above ____.

60 — Correct Option

Solution :
60
Here some of the dependencies are present:



So, number of possibilities are :
1. 2 choices for 33 and 24 either 33 then 44 or 44 then 33.
2. After that 23 will be come.
3. After that 64 will come.

Now, here 91 and 77 can come in any order, i.e., 5 × 6.
So, total choices will be = 2 × 5 × 6 = 60

QUESTION ANALYTICS +

Q. 14 Solution Video | Have any Doubt ?

A message is made up entirely of characters from the set P = {*W, X, Y, Z*}. The table of probability for each characters given below:

Character	Probability
W	0.01

X	0.30
Y	0.34
Z	0.35

Which of the following are correct?

A If a message of 200 characters over P are encoded using Huffman encoding then the expected length of the encoded message is 340 bits.

B If a message of 200 characters over P are encoded using Huffman encoding then the expected length of the encoded message is 392 bits. Correct Option

C *W* and *X* having the same number of encoding bits when encoded using Huffman coding. Correct Option

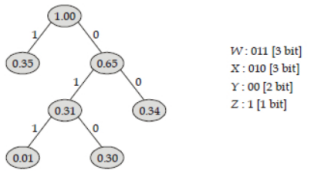
D X and Y having the same number of encoding bits when encoded using Huffman coding.

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

Solution :

(b, c)

Using min heap data structure:



Expected length : $[0.35 \times 1 + 0.34 \times 2 + 0.30 \times 3 + 0.01 \times 3] \times 200$
$= [0.35 + 0.68 + 0.90 + 0.03] \times 200$
$= [1.96] \times 200 = 392$

QUESTION ANALYTICS +

Q. 15 ? FAQ ▶ Solution Video Have any Doubt ?

Consider the following function written in a C like language:

```
int Bar (int n)
{
    if (n < 2) return;
    else
    {
        int sum = 0;
        int i, j;
        for (i = 1; i <=4; i++) Bar (n/2);
        for (i = 1; i <= n; i++)
        {
            for (j = 1; j <= i; j++)
            {
                sum = sum + 1;
            }
        }
    }
}
```

Which of the below statements are correct?

A The time complexity of Bar (n) is $\theta(n^2 \log(n))$. Correct Option

B The time complexity of Bar (n) is $\Omega(n^2 \log(n^2))$. Correct Option

C The time complexity of Bar (n) is $O(n^3 \log(n^2))$. Correct Option

D None of these

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

Solution :

(a, b, c)

Recurrence relation for Bar (n):

$$T(n) = 4T\left(\frac{n}{2}\right) + \theta(n^2)$$

$\Rightarrow \quad T(n) = \theta(n^2 \log n)$

(a) is correct.

Now, $n^2 \log(n^2) = n^2 . 2 \log n = \Omega(n^2 \log n)$

Hence, $T(n)$ can also be written as

$$T(n) = \Omega(n^2 \log n)$$

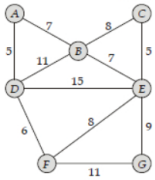
$\therefore$ (b) is correct.

Similarly, (c) is also correct.

QUESTION ANALYTICS +

Q. 16 ▶ Solution Video Have any Doubt ?

Consider the following graph G:



Which of the following options are correct?

A The sum of the weights of an spanning free formed using Kruskal's algorithm on G is 39. Correct Option

B The set of edges in the spanning of G tree using Kruskal's algorithm are (A, D), (C, E), (D, B), (A, B), (B, E) and (G, E).

C The set of edges in the spanning free of G using Kruskal's algorithm are (A, D), (C, E), (D, F), (A, B), (B, E) and (G, E). Correct Option

D The sum of the weights of the spanning free formed using Kruskal's algorithm on G is 35.

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

Solution :
(a, c)
MST of G will have AD, CE, DF, AB, BE, GE as edges. Hence, sum will be
(5 + 5 + 6 + 7 + 7 + 9) = 39

QUESTION ANALYTICS +

Q. 17 Solution Video | Have any Doubt ?

Consider the following pseudo code:

```
for(i = 1; i < m + 1; i++)
for(J = 1; J < m + 1; J+ = i)
temp++;
```

Which of the following is correct time complexity of the above code?

A O(log n)

B O(n)

C O(n log n) Correct Option

Solution :
(c)

$$T(n) = n + \frac{n}{2} + \frac{n}{3} + \ldots\ldots + 1$$
$$= n\left(1 + \frac{1}{2} + \frac{1}{3} + \ldots\ldots + \frac{1}{n}\right)$$
$$= O(n \log n)$$

D O(n^2)

QUESTION ANALYTICS +

Q. 18 FAQ | Solution Video | Have any Doubt ?

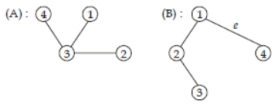
Consider $G = (V, E)$ be a connected graph with n vertices and m edges with distinct positive edge weight. $T = (V, E)$ be a spanning tree of G and bottleneck edge of T is the edge with greatest cost in T. A spanning tree T of G is a minimum bottleneck spanning tree if there is no spanning tree T' of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

A Only S_1

B Only S_2

C Both S_1 and S_2 Correct Option

Solution :
(c)
S_1 is true
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 heaviest. 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.

D Neither S_1 nor S_2

QUESTION ANALYTICS +

Q. 19 ? FAQ ▶ Solution Video Have any Doubt ?

Which of the following represents the number of elements that can be sorted in $\theta(n)$ times using merge sort?

A $\theta(\log n)$

B $\theta(n)$

C $\theta\left(\frac{n}{\log n}\right)$ Correct Option

Solution :
(c)
Time complexity to sort n elements using merge sort = $\theta(n \log n)$

$$\theta(n) = \theta\left(\frac{n}{\log n}\log\frac{n}{\log n}\right)$$

$$\theta(n) = \theta\left(\frac{n}{\log n}[\log n - \log\log n]\right)$$

$$\theta(n) = \theta\left(\frac{n}{\log n}\log n\right) [\log n - \log\log n = O(\log n)]$$

$$\theta(n) = \theta(n)$$

D $\theta(\sqrt{n})$

QUESTION ANALYTICS +

Q. 20 ? FAQ ▶ Solution Video Have any Doubt ?

Consider an array containing 'n' elements. The elements present in an array are in arithmetic progression, but one element is missing in that order. What is the time complexity to find the position of the missing element using divide and conquer?

A $O(n)$

B $O(n^2)$

C $O(\log n)$ Correct Option

Solution :
(c)
The time complexity is $O(\log n)$ using binary search.

The idea is to go to the middle element at index $\frac{n}{2}$ calculate $a_{n/2} = a + \left[\frac{n}{2} - 1\right] \times d$ and check

$a\left[\frac{n}{2}\right] = a_{n/2}$ or not if equal check on RHS only otherwise LHS.

D $O(n \log n)$

QUESTION ANALYTICS +

Kunal Jha
Course: GATE
Computer Science Engineering(CS)

ALGORITHMS (GATE - 2021) - REPORTS

OVERALL ANALYSIS | COMPARISON REPORT | SOLUTION REPORT

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

Q. 21 FAQ | Solution Video | Have any Doubt ?

Consider a procedure find ( ) which take array of n integers as input and produce pair of elements of array whose difference is not greater than the difference of any other pair of element of that array. Which of the following represent worst case time complexity of find ( ) procedure?

A. $O(n)$

B. $O(n \log n)$ — Correct Option

Solution :
(b)
Using divide and conquer approach, closest pair can be found in $O(n \log n)$ time.
Algorithm :
Step 1 : Divide the set into two equal sized parts by the line l, and recursively compute the distance in each part. [d_1 = closest pair (left half); d_2 = closest pair (right half)] and returning the points in each set in order that is sorted by y-coordinate].
Step 2 : Let 'd' be the minimal of two minimal distances

$$d = \min(d_1, d_2) \ldots O(1)$$

Step 3 : Eliminate points that lie farther than 'd' apart from l, ... $O(n)$.
Step 4 : Merge the two sorted lists into one sorted list .... $O(n)$.
Step 5 : Scan the remaining points in the y-order and compute the distances of each point to its 5 neighbour ... $O(n)$.
Step 6 : If any of these distances is less than 'd' the update 'd' ... $O(1)$.

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

$$T(n) = O(n \log n)$$

C. $O(n^2)$

D. $O(n^2 \log n)$

QUESTION ANALYTICS +

Q. 22 FAQ | Solution Video | Have any Doubt ?

We are given log m sorted lists, each size $\frac{\log n}{\log m}$. The time complexity of merging the lists into a single sorted list using merge sort is equal to

A. $O(\log n . \log \log m)$ — Correct Option

Solution :
(a)
First let's find the height of the tree (say h).

$$\frac{(\log m)}{2^h} = 1$$

$$h = O(\log \log m)$$

The time to merge from level i to level $i + 1 = O(\log n)$.
So, the total time to merge log m sorted lists into a single list of log n elements

$$= O(\log n . \log \log m)$$

B. $O(\log m . \log \log n)$

C. $O(\log m \log n)$

D. $O(m \log \log n)$

QUESTION ANALYTICS +

Q. 23 Solution Video | Have any Doubt ?

Match **List-I** with **List-II** and select the correct answer using the codes given below the lists:

List-I	List-II
A. $T(n) = T(n-1) + n$	1. $O(n \log n)$
B. $T(n) = T\left(\frac{n}{2}\right) + 1$	2. $O(n)$
C. $T(n) = 2T\left(\frac{n}{2}\right) + n$	3. $O(\log n)$
D. $T(n) = 2T\left(\frac{n}{2} + 8\right) + n$	4. $O(n^2)$

Codes:

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

A a

B b

C c

D d — Correct Option

Solution :
(d)

A. $T(n) = T(n-1) + n \Rightarrow T(n) = O(n^2)$

B. $T(n) = T\left(\frac{n}{2}\right) + 1 \Rightarrow T(n) = O(\log n)$

C. $T(n) = 2T\left(\frac{n}{2}\right) + n \Rightarrow T(n) = O(n \log n)$

D. $T(n) = 2T\left(\frac{n}{2} + 8\right) + n \Rightarrow T(n) = O(n \log n)$

Therefore, option (d) is correct.

QUESTION ANALYTICS +

Q. 24 Solution Video | Have any Doubt ?

What is the time complexity of best known algorithm for reversing a singly linked list in a group of given size k for example if the given linked list is 1 → 2 → 3 → 4 → 5 → 6 → 7 → 8 → NULL and *k* is 3 then output should be 3 → 2 → 1 → 6 → 5 → 4 → 8 → 7 → NULL? (where n is the number of node in the linked list).

A $O(n)$ — Correct Option

Solution :
(a)

To reverse the singly linked list in a group of given size k.

1. Reverse the first sub-list of size k while reversing keep track of the next node and previous node. Let the pointer to the next node be next and pointer to the previous node be previous.
2. Head → next = reverse (next, k) (recursively call for rest of the list and link the two sublists).
3. return prev (prev becomes the new head of the list)

```
while (current! = NULL && count < k)
{
    next = current → next;
    current → next = prev;
      prev = current;
      current = next;
      count ++;
}  /* reverse first k nodes */
    if (next! = NULL)
    head → next = reverse (next, k);
    return prev;
}
```

Here every node is processed just once, time complexity of the function is $O(n)$. So, correct option is (*a*).

B $O(k.n)$

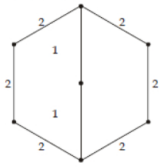
C $O(n^2)$

D $O(n \log k)$

QUESTION ANALYTICS +

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

Consider the following graph in which x represent the number of minimum cost spanning tree and y represent the number of second minimum cost spanning tree (minimum spanning tree with cost less than all other minimum spanning tree except actual minimum cost spanning tree). What is the value represents $|2^x - 2^y|$ where || represent mod value? _____.



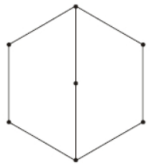
3584 — Correct Option

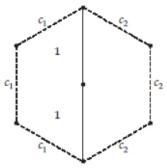
Solution :
3584

Three choices from c_1 and three choices from c_2

$\Rightarrow \quad 3c_2 \times 3c_2 = 9$

So, $\quad x = 9$





We want second minimum spanning tree so for middle point we have 2 choices and rest there are 6 edges.

$\Rightarrow \quad 6c_5 \times 2c_1 = 12$

So, $\quad y = 12$

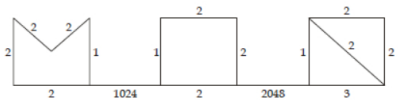
$|2^9 - 2^{12}| = 2^9 \times (1 - 2^3)$

$= 2^9 \times 7 = 3584$

QUESTION ANALYTICS +

Q. 26 Solution Video | Have any Doubt ? | Bookmark

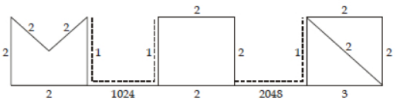
The number of MSTs for the graph given below are _____.



Graph 'H'

36 — Correct Option

Solution :
36



(Dotted edge are those included in every MST)

Number of MSTs $= ({}^4C_3 \times {}^3C_2 \times {}^3C_2)$

$= (4 \times 3 \times 3) = 36$

QUESTION ANALYTICS +

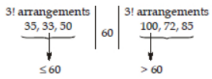
Q. 27 FAQ | Solution Video | Have any Doubt ? | Bookmark

Consider the following array *A*. Quicksort is run on the array *A* and assume that the algorithm picks the first element as pivot. In how many ways can the elements present in the array be arranged so that the effect of first pass of quicksort algorithm is preserved?
A = (60, 50, 100, 35, 33, 72, 85)

36 — Correct Option

Solution :
36

After running first pass of quicksort, the elements will get divided into partitions.



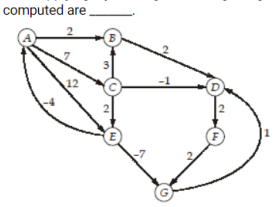
(Number of arrangements preserving first pass of quicksort)

$= 3!\,3! = 36$

QUESTION ANALYTICS +

Q. 28 FAQ | Solution Video | Have any Doubt ? | Bookmark

While applying Dijkstra algorithm on given graph starting from vertex A, it is already known that it is computing wrong path to some of the vertices. The number of wrong path computed are _____.



2 Correct Option

Solution :
2

	Using Dijkstra algorithm		Actual Shortest path	
$A \to B$	2	AB	2	AB
C	7	AC	7	AC
D	4	ABD	3	$ACEGD$
E	9	ACE	9	ACE
F	6	$ABDF$	5	$ACEGDF$

So, there are 2 wrong path calculated to D, F vertices.

QUESTION ANALYTICS +

Q. 29 FAQ Solution Video Have any Doubt ?

Consider two persons (person X, person Y). Person X who was given a problem to calculate $A_1 \times A_2 \times A_3$ with dimension 3×100, 100×2 and 2×2 in minimum multiplication. Person X knows only Greedy algorithm (multiply matrix which gives less number of multiplication) and solve $A_1 \times A_2 \times A_3$ with M_1 multiplications. Person Y solved the same problem using Dynamic algorithm with M_2 multiplications. How many number of multiplications saved by person Y than person X? _____.

388 Correct Option

Solution :
388

$$A_1A_2A_3 = A_1 \times (A_2 \times A_3)$$

$3 \times 100, 100 \times 2, 2 \times 2$

By Person X applying Greedy:

$A_1 \times (A_2 \times A_3)$

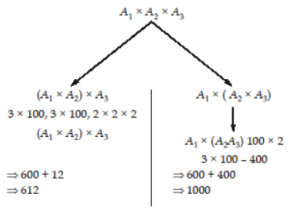
$3 \times 100, 100 \times 2, 2 \times 2$

$(A_2A_3) \to 100 \times 2, 2 \times 2 = 200 \times 2 = 400$

$A_1 \times (A_2A_3) \to 3 \times 100, 100 \times 2 = 300 \times 2 = 600$

Total number of multiplication required = 600 + 400 = 1000

Person Y with Dynamic :



Number of multiplication saved by Person Y = 1000 – 612 = 388

QUESTION ANALYTICS +

Q. 30 Solution Video Have any Doubt ?

Assume we are given 2 algorithms A_1 and A_2. For an input size 'n', algorithm A_1 takes $\theta(n)$ time and A_2 takes $\theta(n^2)$ time respectively. For $n = 16$, algorithm A_1 takes 48 seconds; while A_2 takes 512 seconds respectively. Let $T(A_1)$ and $T(A_2)$ denote the time taken by A_1 and A_2 when made to run on an input size $n = 64$ respectively. Then the value of $T(A_2) - T(A_1)$ will be equal to _____.

8000 Correct Option

Solution :
8000

Algorithm (A_1):

For $n = 16$, A_1 takes 48 seconds

$$\Rightarrow \quad cn = 48$$
$$c(16) = 48 \Rightarrow c = 3$$

So, for $n = 64$, $T(A_1)$ will equal to,

$$T(A_1) = c.n = 3(64) = 192 \text{ seconds}$$

Algorithm (A_2):

For $n = 16$, 512 seconds

$$kn^2 = 512$$
$$\text{or} \quad k(16)^2 = 512$$
$$\therefore \quad k = 2$$

Now, for $n = 64$

$$T(A_2) = 2 \times (64)^2 = 2 \times 4096 = 8192 \text{ seconds}$$
$$\text{Therefore,} \quad T(A_2) - T(A_1) = (8192 - 192) \text{ seconds}$$
$$= 8000 \text{ seconds}$$

QUESTION ANALYTICS +

Kunal Jha
Course: GATE
Computer Science Engineering(CS)

ALGORITHMS (GATE - 2021) - REPORTS

OVERALL ANALYSIS | COMPARISON REPORT | SOLUTION REPORT

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

Q. 31 Solution Video | Have any Doubt ?

Which of the following statements are true?

A. The depth of any DFS free rooted at a vertex is at least as much as the depth of any BFS tree rooted at the same vertex. — Correct Option

B. For a directed graph, the absence of back edges in a DFS free means graph has no cycle. — Correct Option

C. BFS takes $O(V^2)$ time in a graph G(V, E) if graph is represented with an adjacency matrix. — Correct Option

D. If all the edges in a graph have distinct weight then the shortest path between two vertices is unique.

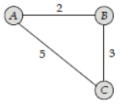
YOUR ANSWER - NA | CORRECT ANSWER - a,b,c | STATUS - SKIPPED

Solution :

(a, b, c)

- Only option (d) is false because distance 2 vertices can have different paths.

Example:



All edges having distinct weight. Here A to C have two different paths both having same cost of 5 unit.

QUESTION ANALYTICS +

Q. 32 FAQ | Solution Video | Have any Doubt ?

Consider the following statements and select the incorrect options.

A. An undirected graph is said to be Hamiltonian if it has a cycle containing all the vertices. Any DFS tree on a Hamiltonian graph must have depth V-1.

B. For all weighted graphs, for any two vertices 's' and 't'. Bellman at 's' will always return a shortest path to 't'. — Correct Option

C. 'V' innovation of Dijkstra's algorithm i.e., O(V * E log V) is best for all pairs shortest path problem. (V = Number of vertices and E = Number of edges). — Correct Option

D. '1' innovation of Floyd-Warshall algorithm i.e., O(V3) is the best algorithm for all pair shortest path problem.

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

Solution :

(b, c)
Option (b) and (c) are incorrect.

QUESTION ANALYTICS +

Q. 33 Solution Video | Have any Doubt ?

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

Each character takes 1 bits. If compression technique used is Huffman coding, then which of the following are correct?

A. The average length of the encoded message is 3.12 bits/char.

B. The average length of the encoded message is 2.56 bits/char. — Correct Option

C. Letter *b* and *c* required 3 bit and 2 bit respectively for represented in Huffman coding. — Correct Option

D. Letter *d* and *e* required 2 bit and 3 bit respectively for represented in Huffman coding. — Correct Option

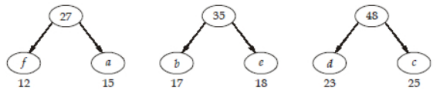
YOUR ANSWER - NA | CORRECT ANSWER - b,c,d | STATUS - SKIPPED

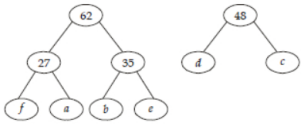
Solution :

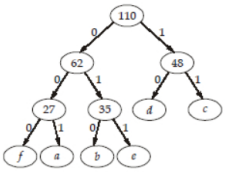
(b, c, d)

Total characters are 110 bits.

As Huffman tree:







According to Huffman encoding

$a = 001 \Rightarrow 3$ bit

$b = 010 \Rightarrow 3$ bit

$c = 11 \Rightarrow 2$ bit

$d = 10 \Rightarrow 2$ bit

$e = 011 \Rightarrow 3$ bit

$f = 000 \Rightarrow 3$ bit

Number of bits needed $= (3 \times 12) + (3 \times 15) + (3 \times 17) + (3 \times 18) + (2 \times 23) + (2 \times 25) = 282$

$$\text{Average bits} = \frac{282}{110} = 2.56 \text{ bits/char}$$

QUESTION ANALYTICS +