Sem IV IT

MSE Synoptic

March 2020

Design and Analysis of Algorithm

Q. No.	Questions	Max. Marks

Q1 a. Analyze the asymptotic time complexity of given fun(). Justify your answer. — **02**

Answer: O(n)-------------------- 1 mark

Justification:------------------1 mark

For a input integer n, the innermost statement of fun() is executed following times.

$$n + n/2 + n/4 + \dots 1$$

So time complexity T(n) can be written as

$T(n) = O(n + n/2 + n/4 + \dots 1) = O(n)$

The value of count is also $n + n/2 + n/4 + .. + 1$

b. — **02**

Justification : $n \sqrt{n}$ is not $O(n^2 \log n)$ so (A) is false. Both (B) and (C) are true.----1 mark

Accepted Answers:

(B) and (C) are true but (A) is not true.-----1 mark

c. **Answer:** $T(n) = \Theta(n^2)$ (Case 3) — **02**

Marks Distribution:

Final correct answer with correct Case of MT applicable mentioned -----------1 mark

Correctly shown the master theorem applicability-------------------------------1 mark

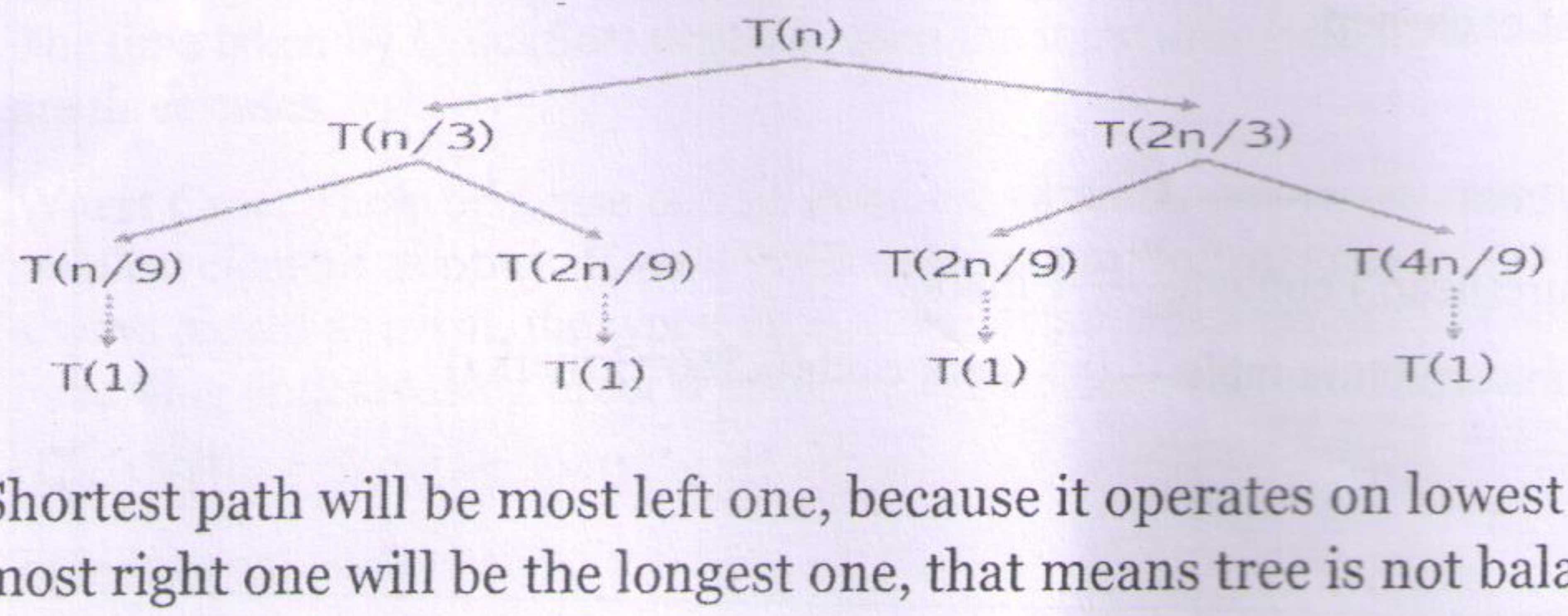
OR

Marks distribution:

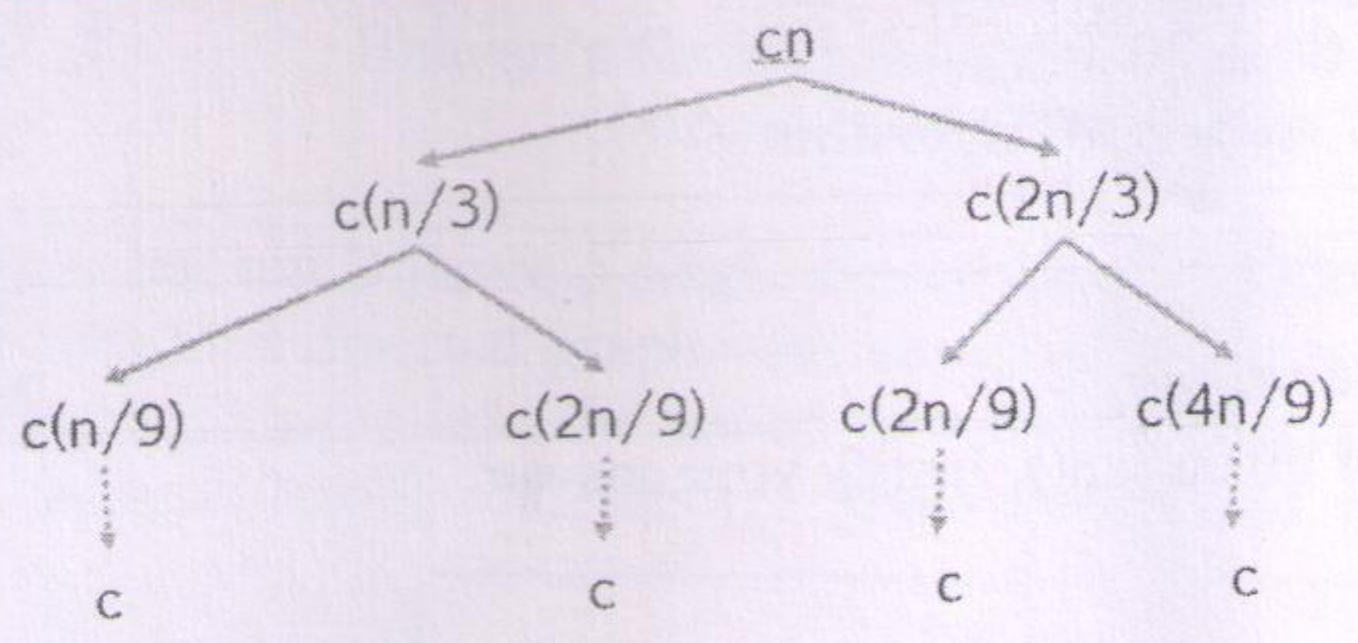
Tree drawn correctly ---------------1 mark

Sum of each level shown correctly-----0.5 mark

Final answer --------------0.5 mark



Shortest path will be most left one, because it operates on lowest value, and the most right one will be the longest one, that means tree is not balanced.

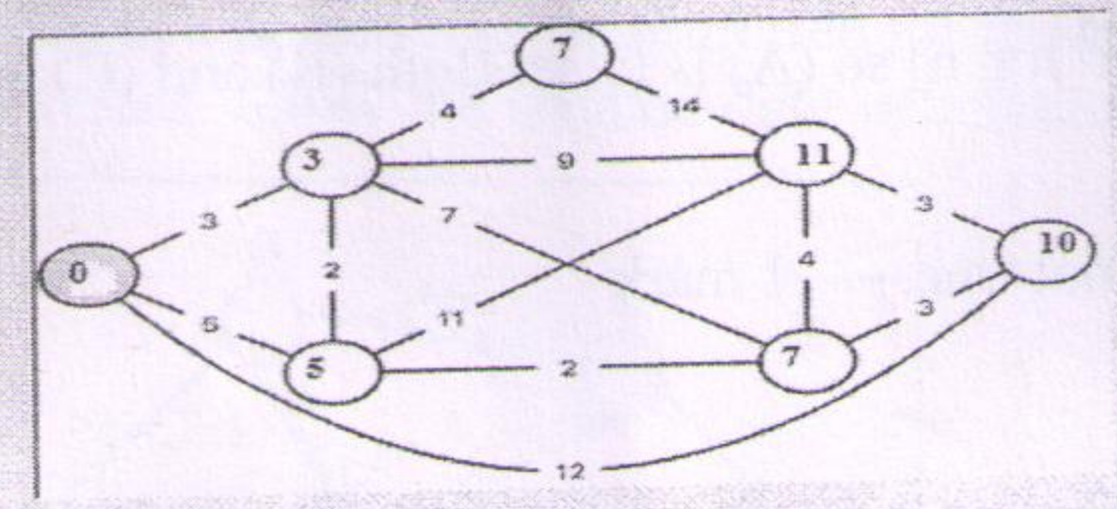


Elements from shortest path are being divided by 3, so length of this path will be equal to $\log_3 n\log_3 n$. So if number of complete levels of recursion tree for shortest path is equal to $\log_3 n$, that means cost of algorithm for this path will be: **$T(n)=cn\log_3 n=\Omega(n\lg n)$**

Q2

Final SSS Graph with distance:-----------------------------1 mark

05



final updated table:----------------------------------1 mark

VERTEX	A	B	C	D	E	F	G
KEY	0	3	5	11	7	10	7
PARENT	----	A	A	E	C	E	B

6 steps of updating the data structure table----0.5 mark each(0.5*6=3 marks)

OR

Using Prim's algorithm construct a minimum spanning tree for the given graph. Show the updated data structured at each step .

Marks Distribution:

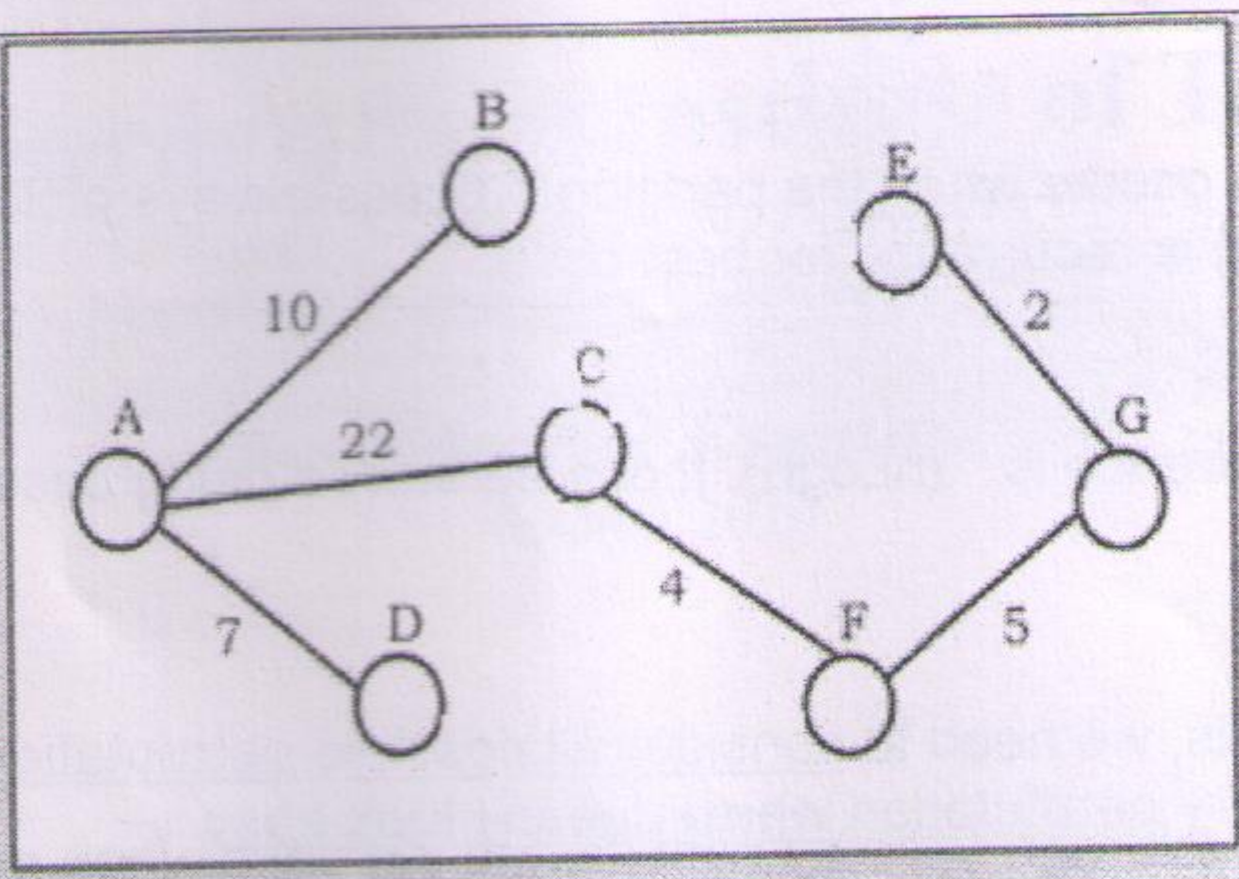
Correct MST graph with cost ---------------1 marks

Correct final Data structure(heap) content------1 mark

6 steps of updating the data structure table----0.5 mark each(0.5*6=3 marks)

Final Correct answer :

MST Cost=50



final content of data structure

VERTEX	A	B	C	D	E	F	G
KEY	0	10	22	7	2	4	5
PARENT	----	A	A	A	G	C	F

Q4. Marks Distribution : **05**

For each case following is required as a part of answer

1. Recurrence relation
2. Recursion tree
3. Time complexity
4. Example of input test case

Best Case ---------1 mark

Worst case -------2 marks

Average Case ------2 marks

Answer:

Analysis of QuickSort

Time taken by QuickSort in general can be written as following.

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

The first two terms are for two recursive calls, the last term is for the partition process. k is the number of elements which are smaller than pivot.

The time taken by QuickSort depends upon the input array and partition strategy. Following are three cases.

Worst Case: The worst case occurs when the partition process always picks greatest or smallest element as pivot. If we consider above partition strategy where last element is always picked as pivot, the worst case would occur when the array is already sorted in increasing or decreasing order. Following is recurrence for worst case.

T(n) = T(0) + T(n-1) + (n)

which is equivalent to

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

The solution of above recurrence is (n^2)

Best Case: The best case occurs when the partition process always picks the middle element as pivot. Following is recurrence for best case.

```
T(n) = 2T(n/2) +   (n)
```

The solution of above recurrence is   (nLogn). It can be solved using case 2 of Master Theorem.

Average Case:
To do average case analysis, we need to consider all possible permutation of array and calculate time taken by every permutation which doesn't look easy.
We can get an idea of average case by considering the case when partition puts O(n/9) elements in one set and O(9n/10) elements in other set. Following is recurrence for this case.

```
T(n) = T(n/9) + T(9n/10) +   (n)
```

Solution of above recurrence is also O(nLogn)

Although the worst case time complexity of QuickSort is $O(n^2)$ which is more than many other sorting algorithms like Merge Sort and Heap Sort, QuickSort is faster in practice, because its inner loop can be efficiently implemented on most architectures, and in most real-world data. QuickSort can be implemented in different ways by changing the choice of pivot, so that the worst case rarely occurs for a given type of data. However, merge sort is generally considered better when data is huge and stored in external storage.

Q5

7 Formulas each formula P-V -----------0.5 marks each (0.5 *7=3.5)

Final Correct answer--------------- 0.5 marks

04

$$A = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} \quad B = \begin{bmatrix} 5 & 6 \\ 7 & 8 \end{bmatrix}$$

$$\begin{aligned}
P &= (A_{11} + A_{22})(B_{11} + B_{22}) \\
&= (1 + 4)(5 + 8) = 5 \times 13 = 65 \\
Q &= (A_{21} + A_{22})B_{11} \\
&= (3 + 4) \times 5 = 7 \times 5 = 35 \\
R &= A_{11}(B_{12} - B_{22}) \\
&= 1 \times (6 - 8) = 1 \times -2 = -2 \\
S &= A_{22}(B_{21} - B_{11}) \\
&= 4 \times (7 - 5) = 4 \times 2 = 8
\end{aligned}$$

$$\begin{aligned}
T &= (A_{11} + A_{12})B_{22} \\
&= (1 + 2) \times 8 = 3 \times 8 = 24 \\
U &= (A_{21} - A_{11})(B_{11} + B_{12}) \\
&= (3 - 1)(5 + 6) = 2 \times 11 = 22 \\
V &= (A_{12} - A_{22})(B_{21} + B_{22}) \\
&= (2 - 4)(7 + 8) = -2 \times 15 = -30 \\
C_{11} &= P + S - T + V \\
&= 65 + 8 - 24 - 30 = 19 \\
C_{12} &= R + T = -2 + 24 = 22 \\
C_{21} &= Q + T = 35 + 8 = 43 \\
C_{22} &= P + R - Q + V \\
&= 65 - 2 - 35 + 22 = 50
\end{aligned}$$

Thus, the required matrix product is $C = \begin{bmatrix} 19 & 22 \\ 43 & 50 \end{bmatrix}$