North East University Bangladesh

Department of Computer Science and Engineering
Semester Final Examination (Summer 2022)
Program: B.Sc. (Eng.) in CSE
Course: CSE 231 (Algorithm Design and Analysis)

Total Marks: 40 | Time: 2 hours

1. Answer any ***five*** from the following: 5x2=10

   1.1 In algorithmic complexity analysis, ***big-theta*** notation is stronger than ***big-Oh*** notation. True or false, explain.

   1.2 Find the big-oh (O) of this function: $f(n) = 5n \log n + 2n + 1024$

   1.3 Is it possible to find all pairs of shortest paths using Dijkstra's algorithm? Justify.

   1.4 Discuss the similarity between ***Breadth First Search (BFS)*** and ***Dijkstra's*** algorithm.

   1.5 Compare and contrast between ***Greedy*** and ***Dynamic*** programming paradigms.

   1.6 Classify the following algorithms according to their programming approach: Minimum Spanning tree, Merge-sort, Quick-sort, 0/1 knapsack, Floyd-Warshall's shortest-path.

2. Answer any ***three*** from the following: 3x4=12

   2.1 Discuss the elements of dynamic programming.

   2.2 Write an algorithm for the fractional Knapsack problem. Find the time complexity of the algorithm.

   2.3 Consider the following MERGE-SORT procedure, which sorts the array assuming subarrays `A[st...mid]` and `A[mid+1...end]` are already sorted:

```
MERGE-SORT(A, st, end)
1. If st<end, Then
2.     mid ← ⌊(st + end)/2⌋
3.     MERGE-SORT(A,st,mid)
4.     MERGE-SORT(A,mid+1,end)
5.     MERGE(A,st,mid,end)
```

   Describe what *MERGE-SORT* does to `A` if we remove the last line of the algorithm so that we never call *MERGE*.

   2.4 Solve the following activity selection problem using greedy method:

i	1	2	3	4	5	6	7	8	9	10
S_i	3	5	1	4	1	1	6	7	2	10
F_i	6	11	2	7	9	3	8	10	5	12

   Find the complexity of the procedure (count the time required for sorting).

3. Answer any ***three*** from the following: 3x6=18

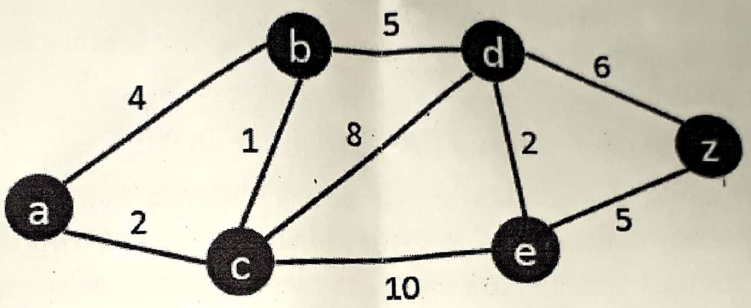
3.1 Solve the following recurrence relations (any two):

a) $T(n) = 8T\left(\frac{n}{2}\right) + n^2$

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

c) $T(n) = 3T\left(\frac{n}{3}\right) + \sqrt{n}$

3.2 Using Dijkstra's algorithm calculate the single-source shortest path for the following graph (use node 'a' as source).



3.3 Find the longest common subsequence of the following two strings X and Y using dynamic programming.

$$X = abbacdcb,\ Y = aabdcba$$

3.4 Find the minimum spanning tree of the following graph using Prim's algorithm.