Bangladesh Army University of Science and Technology

Department of Computer Science and Engineering

Final Examination, Fall 2018 | **Level-2 Term-II**
Course Code: CSE 2201 | **Course Title: Algorithms**
Time: 03 (Three) hours | **Full Marks: 210**

N.B. (i) Answer any three questions from each PART (ii) Use separate answer script for each PART
(iii) Marks allotted are indicated in the margin (iv) Symbols have their usual meanings

PART A

(Answer any **three** questions)

1. a) How do you compare two algorithms for the same problem? **5**

   b) How do you define the correctness of an algorithm? Write the linear search algorithm and prove its correctness. **2+3+5=10**

   c) Suppose you are given n points in the plane and each point is specified by its (x, y) coordinates. Give an algorithm to find the pair of points that are closest together. Analyze the time complexity of the algorithm. **6+4=10**

   d) Give an algorithm for merging two sorted lists in O(n) time. You need to prove the time complexity of your algorithm. **5+5=10**

2. a) Assume that you are given a set of n task. Each task I has a start time S_i and a finish time f_i. You need to schedule the tasks so that all the tasks can be done using the minimum number of machines. Give a greedy algorithm to solve the task scheduling problem mentioned above. Prove the correctness of the algorithms. **5+5=10**

   b) Assume that you want to sort a set of data using the merge sort technique. However, in each recursive step you divide the data set into three parts of almost equal sizes. Write the recurrence relation for your merge sort algorithm and solve the recurrence relation using the recursion tree method. **4+4=8**

   c) Write the recurrence relation for time complexity of binary search and solve it using Master Theorem. **2+5=7**

   d) Algorithm A solves problems of size n by dividing them into seven sub-problems of size n/2, recursively solving each sub-problem, and then combining the solutions in $O(n^2)$ time. Write the recurrence relation for the time complexity of algorithm, A and solve it using Master Theorem or any other method. **4+6=10**

3. a) i) Write down the pseudo code of Quicksort algorithm. **5+5=10**
      ii) In quicksort, what is the best case partitioning and the worst case partitioning? By considering your algorithm, give one sample input for each case.

   b) What are the key properties of an optimization problem such that the problem can be solved using dynamic programming? How do you calculate the running time of an algorithm designed by dynamic programming method? **5+5=10**

   c) Find all longest common subsequences of $X = \{ATGCTGAT\}$ and $Y = \{TGGCATA\}$ using dynamic programming method. **15**

4. a) What is spanning tree? Write Prim's algorithm that computes a minimum spanning tree of a graph. Analyze the time-complexity of the algorithm. **2+6+7=15**

   b) What is Topological sort? Perform topological sort on graph G given in figure 1. You should mention every step. **2+6=8**

   [whenever there is a choice of vertices, always use numeric order]

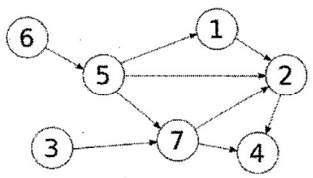

Figure 1: Graph G

c) What do you understated by strongly connected graph? Write an algorithm that determine whether a graph is strongly connected or not. By using your algorithm prove that the graph G shown in figure 1 is not a strongly connected graph. **2+4+6=**

PART B

(Answer any **three** questions)

5. a) i) Can the shortest path of a graph contain any cycle? Justify your answer. **5+5=10**

ii) Prove that, sub-paths of shortest paths are also shortest paths.

b) Consider the given pseudo code for Dijkstra's algorithm. Now, write additional pseudo code for Initialize-Single-Source() and Relax() function. Using all these, find the shortest path between vertex ***a*** and ***z*** of graph H shown in figure 2. You should show each vertex selection in separate figures and corresponding relaxation on the same figure. You should stop after finding the shortest path between s and z (don't need to run for the whole graph) and draw the shortest path between s and z. **5+10=15**

```
DIJKSTRA(G, w, s)
1  INITIALIZE-SINGLE-SOURCE(G, s)
2  S = ∅
3  Q = G.V
4  while Q ≠ ∅
5      u = EXTRACT-MIN(Q)
6      S = S ∪ {u}
7      for each vertex v ∈ G.Adj[u]
8          RELAX(u, v, w)
```

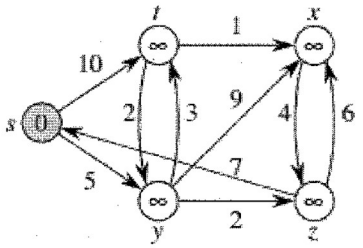

Figure 2:Graph H

c) Show by example that the Dijkstra's algorithm gives wrong result for a graph with negative weight. **5**

d) Suppose, by using Floyd Warshall all pair shortest path algorithm on a graph, we have found the following path matrix and distance matrix. Now, find the minimum distance between A and D. Also mention the path as a sequence of vertices. **5**

	A	B	C	D
A	A	A	B	C
B	D	B	B	C
C	D	A	C	C
D	D	A	B	D

Path matrix

	A	B	C	D
A	0	3	1	3
B	1	0	-2	0
C	3	6	0	2
D	1	4	2	0

Distance Matrix

6. a) What is flow network? Write the Edmonds-Karp algorithm that solves the maximum-flow problem. Explain how network flow algorithm allows us to find the maximum bipartite matching. **2+8+5=15**

b) For the flow network N shown in figure 3, find the value of the maximum flow by drawing residual networks in each step. Finally, draw the maximum flow network. Also draw the residual network and identify the min-cut of the maximum flow network. **20**

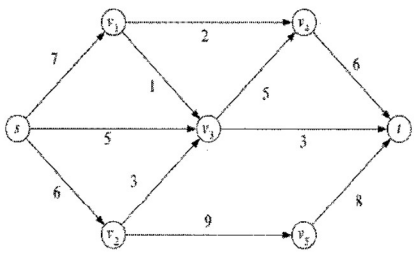

Figure 3: Flow network N

7. a) What do you mean by X<=p Y? Show that the independent set problem and the vertex cover problem is equally hard. **5+10=15**

b) Describe the relationship among P, NP, NP-hard and NP-complete problems with necessary diagrams. **8**

c) Show that the vertex cover problem is in NP-hard by reducing the 3-SAT problem to vertex cover problem. **12**

8. a) Compare backtracking, and branch-and-bound techniques. **5**

b) What is state-space-tree? Solve the given instance of the 0/1 knapsack problem using the branch-and-bound approach with a state-space-tree. Assume the knapsack capacity is 15. **3+12=15**

Item	Weight	Value
1	7	140
2	6	132
3		90
4	3	140

c) Give an approximation algorithm for the vertex cover problem. Analyze the approximation ratio of your algorithm. **15**