# International Islamic University Chittagong

## Department of Computer Science and Engineering

*B. Sc. in CSE Final Examination, Autumn 2018*

**Course Code: CSE 2403 Course Title: Computer Algorithms**

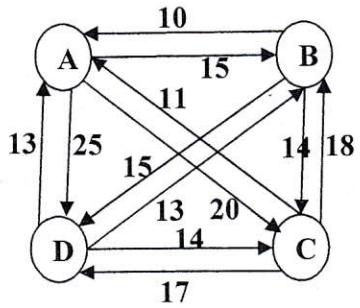
*Total marks: **50** Time: **2** hours **30** minutes*

[Answer any ***two*** questions from **Group-A** and any ***three*** questions from **Group-B**;
Separate answer script must be used for Group-A and Group-B;
Figures in the right hand margin indicate full marks.]

## Group-A

**1.**

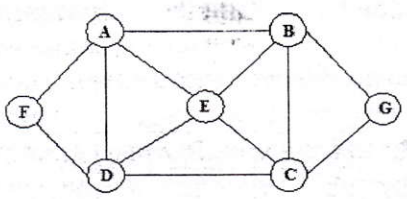
**a)** Define traveling salesperson problem. Apply the traveling salesperson problem on the following figure to find the optimal tour. **4**



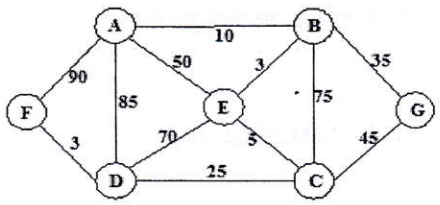

**b)** What is an optimal Huffman code for the following set of frequencies? **3**
a:5 b:8 c:45 d:25 e:17 f:30

**c)** Prove that in activity selection problem, the activity that finishes first is always part of some optimal solution. **3**

**2.** a) Show how Depth First Search works on following graph if E is the source **3**



**b)** Define spanning tree and minimum spanning tree. Construct minimum spanning tree from the graph using Kruskal Algorithm if E is the source. **4**




**c)** In what order do the vertices will be visited if the graph in **Figure 1** is visited using BFS and DFS. Start from vertex ***G*** and assume that the adjacency list is in lexicographic order. You only need to name the vertices. **3**

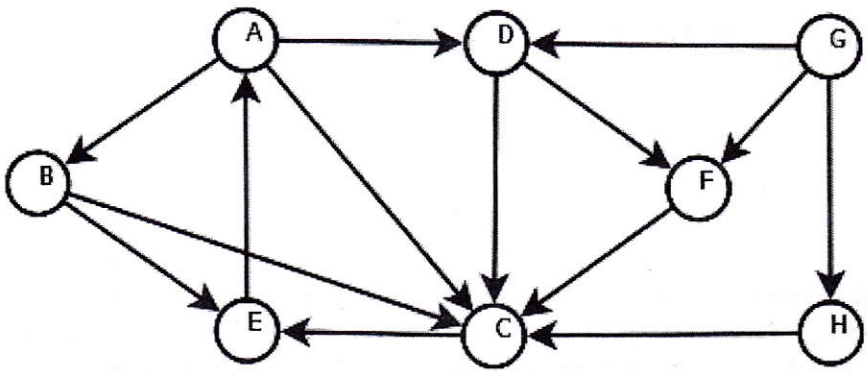



**Figure 1**

**3.**

**a)** Consider the graph with the following weights: w(a, b) = 1, w(a, c) = 2 , w(a, d) = 6, w(a, e) = 9, w(b,c)= 3, w(b, d) = 4, w(b, e) = 10, w(c, d) = 5, w(c, e) = 8, w(d, e) = 7. **5**

a) What minimum spanning tree would *Kruskal*'s algorithm produce? Write the edges in the order that the algorithm would add them to its result.

b) What minimum spanning tree would *Prim*'s algorithm produce, starting at vertex b? Write the edges in the order that the algorithm would add them to its result.

**b)** Describe activity-selection problem. Find all possible optimal set for the following data set using activity selection problem: **5**

| i | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 |
|---|---|---|---|---|---|---|---|---|---|---|---|
| $S_i$ | 1 | 3 | 0 | 5 | 3 | 5 | 6 | 8 | 8 | 2 | 12 |
| $f_i$ | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 |

## Group-B

**4.** a)What are the four variants of shortest path algorithm? Please describe. **3**

**b)** Illustrate the operation of *Dijkstra's algorithm* for finding shortest path on the directed graph shown in **Figure 2**. Assume **a** as the source. Write down the **d** and **π** value of each vertex after each iteration. **5**

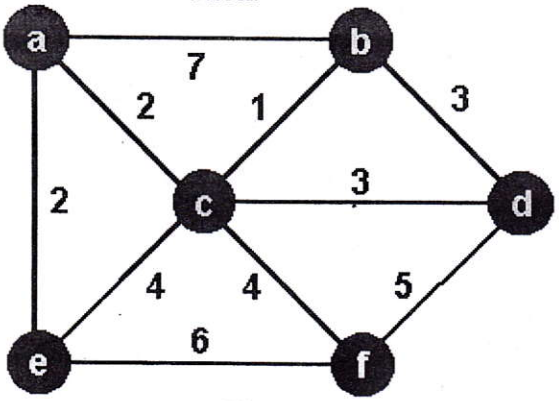



**Figure 2**

**c)** What is *relaxation* of a vertex? Briefly discuss with necessary figure. **2**

**5.**

**a)** Using suitable example show how *branch-and-bound* technique can be applied to solve *travelling salesman problem.* **4**

**b)** What is the significance of **NP-Complete** class in complexity theory? **3**

**c)** Use *recurrence tree* to solve the following recurrence $T(n) = 3T(n/3) + cn^2$. **3**

**6.**

**a)** Write down Graham's scan algorithm of convex hull construction **3**

**b)** Write down the algorithm to calculate $x^n$ mod $m$. **4**

**c)** Graham's scan solves the convex hull problem by sorting the points using polar angle with respect to some point $\mathbf{P_0}$ in counter clockwise. Suppose, you are given two points $\mathbf{P_1(4,5)}$ and $\mathbf{P_2(2,3)}$. Determine which one has greater polar angle with respect to point $\mathbf{P_0(0,0)}$. **3**

**7.**

**a)** What are the applications of Branch and Bound? **4**

**b)** Define convex hull. Consider the following points and find convex hull using Graham's scan algorithm: **3**
P1(0.7, 2.7), P2(1.8, 3.2), P3(2.6, 0.8), P4(0.9, 2.5), P5(0.6, 0.7), P6(1.5, 0.6), P7(1.5, 2.5), P8(2, 1.5), P9(1.0, 2.0), P10(0.8, 1.5).

**c)** State for each of the following algorithms whether they are *greedy algorithm* or *dynamic programming (DP) algorithm* or *neither greedy nor DP.* **3**

i) Bellman-Ford's algorithm for single-source shortest path
ii) Dijkstra's algorithm for single-source shortest path
iii) Floyd-Warshall's algorithm for all-pairs shortest path
iv) Graham's scan algorithm for finding convex hull
v) Kruskal's algorithm for finding minimum spanning tree
vi) Huffman's algorithm for finding optimal prefix code