CSE317 Design & Analysis of Algorithms (Spring'19)

Midterm Examination

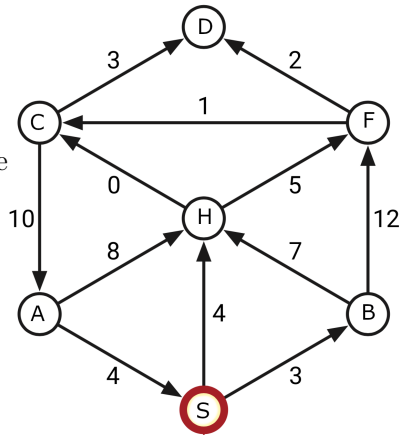
Max Marks: 30 *Time Allowed:* 2 hours

Please give clear and rigorous answers.
Be to the point. Show your work.

Name: _____ **ERP:** _____

Question 1: ..... *5 marks*

Suppose Dijkstra's algorithm is run on the following graph, starting at node S.



(a) Draw a table showing the intermediate Dijkstra scores of all the nodes at each iteration of the algorithm.

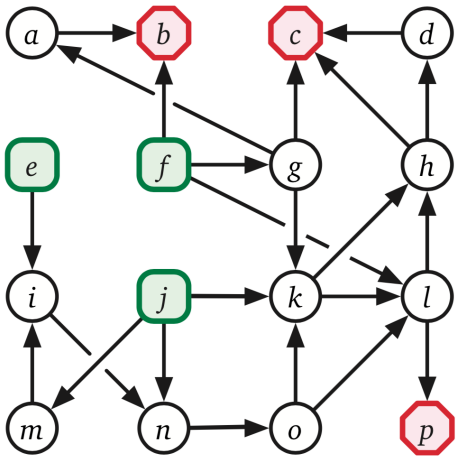
(b) Show the final shortest-path tree.

Solution:

Iteration	**Node removed**	**dist[.] / prev[.] value**						
		S	A	B	C	D	F	H
init	-	0 / nill	∞ / nill	∞ / nill	∞ /nill	∞ /nill	∞ /nill	∞ /nill
1	S	"	"	3 / S	"	"	"	4 / S
2	B	"	"	"	"	"	15 / B	"
3	H	"	"	"	4 / H	"	9 / H	"
4	C	"	14 / C	"	"	7 / C	"	"
5	D	"	"	"	"	"	"	"
6	F	"	"	"	"	"	"	"
7	A	"	"	"	"	"	"	"

Question 2: ..... *5 marks*

Consider the graph G shown on the right.



(a) [1 mark] List all the sources and sinks in G.

(b) [4 marks] Find the topological ordering of G using the DFS-based algorithm discussed in the lectures.

Solution: Sources: e, f, j Sinks: b, c, p

Question 3: ..... *2 marks*

What value is returned by the following function? Express your answer as a function of n. Also give the worst-case running time using the big-O notation.

```
function MYSTERY(n)
    r := 0
    for i := 1 to n − 1 do
        for j := i + 1 to n do
            r := r + 1
    return r
```

Question 4: ..... *4 marks*

Use recursion tree to solve the following recurrence.

$$A(n) = 2A(n/4) + \sqrt{n}$$

Question 5: ..... *4 marks*

Given an array of $n > 0$ distinct integers, design an $O(\log n)$-time algorithm to find a local minimum. A local minimum in an array is an entry that is smaller than all of its adjacent entries. For example, in the array $[23, 45, 32, 12, 5, 3, 6, 56, 77, 33, 55]$, there are three local minima – 23, 3, and 33.

Question 6: ..... *5 marks*

Your friend Tony is planning to hold a Qawwali evening. He wants to take a picture of all the participants, including himself, but he is quite shy and thus cannot take a picture of a person whom he does not know very well. Since he has only shy friends, everyone at the party is also shy. After thinking hard for a long time, he came up with a seemingly good idea:

- Tony brings a disposable camera to the party.
- Anyone holding the camera can take a picture of someone they know very well, and then pass the camera to that person.
- In order not to waste any film, every person must have their picture taken exactly once.

Although there can be some people Tony does not know very well, he knows completely who knows whom well. Thus, in principle, given a list of all the participants, he can determine whether it is possible to take all the pictures using this idea. But how quickly? Give efficient algorithm to solve Tony's problem.

Question 7: ..... *5 marks*

Let's call a directed graph G *semi-connected* if, for every pair of vertices u and v, either u is reachable from v or v is reachable from u (or both).

(a) [1 mark] Give an example of a dag with a unique source that is not semi-connected.

(b) [2 marks] Describe and analyze an algorithm to determine whether a given directed acyclic graph is semi-connected.

(c) [2 marks] Describe and analyze an algorithm to determine whether an arbitrary directed graph is semi-connected.