CSE317 Design & Analysis of Algorithms

Second Term Examination

Max Marks: 30 | Time Allowed: $1\frac{1}{2}$ hours

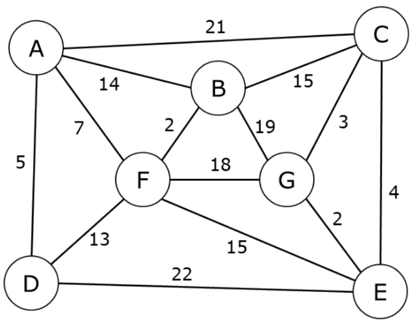
Answer the questions in the spaces provided on the question sheets.
Please give clear and rigorous answers. Be to the point. Show your work.

Name: ____________________ **ERP:** __________

Question:	MSTs	Greedy	Shortest Paths	Divide & Conquer	Total
Marks:	10	6	6	8	30
Score:					

Question 1: MSTs .... *10 marks*

(a) [3 marks] Consider the following undirected graph.



Step through Kruskal's algorithm to calculate a minimum spanning tree of the graph. Show your steps in the table below, including the disjoint sets at each iteration. If you can select two edges with the same weight, select the edge that would come alphabetically last (e.g., select E-F before B-C. Also, select A-F before A-B).

Edge Added	**Edge Cost**	**Running Cost**	**Disjoint Sets**
-	-	0	$\{A\}\{B\}\{C\}\{D\}\{E\}\{F\}\{G\}$

(b) [6 marks] For each of the following statements, either prove it (if it is correct) or give a counterexample (if it isn't correct). Always assume that the graph $G = (V, E)$ is undirected. Do not assume that edge weights are distinct unless this is specifically stated.

i. If G has a cycle with a unique heaviest edge e, then e cannot be part of any MST.

Solution:

ii. Let e be any edge of minimum weight in G. Then e must be part of some MST.

Solution:

iii. If the lightest edge in a graph is unique, then it must be part of every MST.

Solution:

iv. If G has a cycle with a unique lightest edge e, then e must be part of every MST.

Solution:

(c) [1 mark] How long does it take to run Prim's algorithm on a graph $G = (V, E)$? Assume the priority queue is implemented as a heap.

Solution:

Question 2: Greedy .... *6 marks*

(a) [2 marks] You work as a freelancer and have a pool of 10 projects to work on. For each project you know how much money you will get for completing the project. You can complete any 3 projects this month. You want to select such projects that you will get the most money by completing them. What are the safe moves in this problem? (Mark all that apply.)

- ◯ Take the project with the highest payment for completion, complete it and remove it from the pool of projects.
- ◯ If there are more than 3 projects in the pool, remove the project with the lowest payment for completion, don't work on this project. In the other case, remove the first project from the pool and work on this project.
- ◯ Take the project for which you can apply the cool new technology that you've recently learned about.
- ◯ Take the project which you like the most.

(b) [1 mark] In the previous problem, what is the subproblem you need to solve after you've made a safe move?

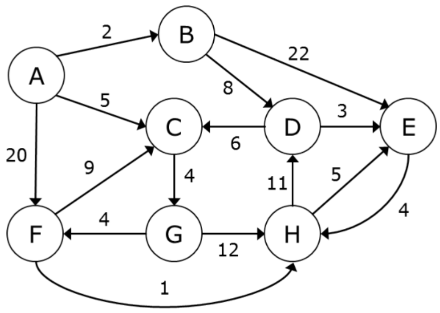
- ◯ Choose projects with highest payment to work on from the pool of projects which now contains only 9 projects.
- ◯ Determine the order in which to work on the selected projects.
- ◯ Compute the sum of payoffs you will get when you complete the selected projects.

(c) [3 marks] Given a set $\{x_1 \leq x_2 \leq \cdots \leq x_n\}$ of points on the real line, determine the smallest set of unit-length closed intervals (e.g. the interval $[1.25, 2.25]$ includes all x_i such that $1.25 \leq x_i \leq 2.25$) that contains all of the points. Give the most efficient algorithm you can to solve this problem, prove it is correct and analyze the time complexity.

Solution:

Question 3: Shortest Paths .... *6 marks*

Consider the following graph.



(a) [4 marks] Use Dijkstra's algorithm to calculate the single-source shortest paths from vertex A to every other vertex. Show your steps in the table below. If during your algorithm two unvisited vertices have the same distance, use alphabetical order to determine which one is selected first.

Iteration	**Node removed**	**`dist[.]` / `prev[.]` value**							
		A	B	C	D	E	F	G	H
0	-	*0* / *nil*	∞ / *nil*	∞ / *nil*	∞ / *nil*	∞ / *nil*	∞ / *nil*	∞ / *nil*	∞ / *nil*
1									
2									
3									
4									
5									
6									
7									
8									

(b) [1 mark] What is the lowest-cost path from A to H in the graph, as computed above?

Solution:

(c) [1 mark] Let $G = (V, E)$ be a weighted graph. Under what assumption does Dijkstra's algorithm correctly compute shortest paths?

Solution:

Question 4: Divide & Conquer .... *8 marks*

(a) [3 marks] Give a $O(\log n)$ algorithm to find the index of last occurrence of an element in a sorted array of size n.

Solution:

(b) [3 marks] *3-way-Merge Sort*: Suppose that instead of dividing in half at each step of Merge Sort, you divide into thirds, sort each third, and finally combine all of them using a three-way merge subroutine. What is the overall asymptotic running time of this algorithm? Give a recurrence describing the running time of 3-way Merge Sort and solve it to justify your answer. (Hint: Note that the merge step can still be implemented in $O(n)$ time.)

Solution:

(c) [2 marks] A "smart" friend claims that he has discovered a data structure which uses only comparisons between elements and supports both `INSERT` and `DELETE-MIN` in $O(1)$ time worst-case. Why is this is impossible?

Solution: