1. [20 points] TRUE/FALSE OR PICK ONE. *No need for justification.*

   (a) TRUE/FALSE
   Let $G = (V, E)$ be a directed graph with weights on edges, and $\gamma(p, q)$ denote the length of the *longest simple path* between p and q. Then, we have the triangle inequality, i.e., $\gamma(p, q) + \gamma(q, r) \le \gamma(p, r)$ for every p, q, and r in V.

   (b) TRUE/FALSE
   Let $G = (V, E)$ be a flow network with source $s \in V$ and sink $t \in V$, and non-negative edge capacities. If the maximum flow assignment for G is *unique* then the minimum cut for this network is also *unique*.

   (c) TRUE/FALSE
   Given two graphs G and G' with the same sets of vertices V and edges E, however different edge weight functions (w and w' respectively). Both weight functions are non-negative and distinct. Moreover, they satisfy the following relation: $w'(e) = w(e)^3$ for every edge e of E. For a pair of vertices u and v in V, a shortest path between them in G is also a shortest path in G'.

   (d) TRUE/FALSE
   Suppose we are given a weighted, directed graph $G = (V, E)$ in which edges that leave the source vertex s may have negative wieghts, all other edge weights are non-negative, and there are no negative-weight cycles. Then, Dijkstra's algorithms correctly finds shortest paths from s in this graph.

   (e) TRUE/FALSE
   Let $G = (V, E)$ be a connected, undirected graph with a weight function $w : E \to \mathbb{N}$ defined on its edges. The shortest path between any two vertices in V is always part of some minimum spanning tree of G.

   (f) TRUE/FALSE
   Let $f_1, f_2 : V \times V \to \mathbb{R}$ be two different flows on a flow network $G = (V, E)$ with a capacity $c : V \times V \to \mathbb{R}^+ \cup \{0\}$. Then, $f_1 + f_2$, which is defined as $(f_1 + f_2)(e) = f_1(e) + f_2(e)$ for every $e \in V \times V$, is also a flow in G.

https://powcoder.com

Assignment Project Exam Help

Add WeChat powcoder

Assignment Project Exam Help

https://powcoder.com

Add WeChat powcoder

(g) True/False

Consider the following pseudocode which describes a greedy algorithm that takes a graph $G = (V, E)$ and a weight function w on its edges as input and returns a set of edges T. The output T of the algorithm is a minimum spanning tree of G.

Maybe-MST(G, w)

Sort the edges of G into non-increasing order of edge weights w

$T \leftarrow E$

for each edge e, taken in non-increasing order by weight

  do if $T - \{e\}$ is a connected graph

    then $T \leftarrow T - e$

return T

(h) Pick One

Let $G = (V, E)$ be a directed graph with edge-weight function $w : E \to \mathbb{R}$. Consider an adjacency matrix $A = (a_{ij})$ where $a_{ij} = w(i, j)$ or ∞. Let $d_{ij}^{(m)}$ denote the weight of a shortest path from i to j that uses at most m edges. Which of the following recurrences correctly formulate a dynamic programming solution for the all-pairs shortest path problem?

(i) $d_{ij}^{(m)} = \min_{i \le k \le j}\{d_{ik}^{(m-1)}\} + a_{kj}$ for $m = 1, 2, \ldots, n - 1$

(ii) $d_{ij}^{(m)} = \min\{d_{ij}^{(m-1)} + a_{ij}\}$ for $m = 1, 2, \ldots, n$

(iii) $d_{ij}^{(m)} = \min_m\{d_{ik}^{(m-1)} + a_{kj}\}$ for $k = 1, 2, \ldots, n$

(iv) $d_{ij}^{(m)} = \min_{1 \le k \le n}\{d_{ik}^{(m-1)} + a_{kj}\}$ for $m = 1, 2, \ldots, n - 1$

(i) Pick One

Let $f_1, f_2 : V \times V \to \mathbb{R}$ be two different flows on a flow network $G = (V, E)$ with a capacity $c : V \times V \to \mathbb{R}^+ \cup \{0\}$. Then, $f_1 + f_2$, which is defined as $(f_1 + f_2)(e) = f_1(e) + f_2(e)$ for every $e \in V \times V$, is NOT necessarily a flow in G because it can violate

(i) the conservation law

(ii) the skew symmetry property

(iii) the capacity constraint

(iv) all of the above

(j) Pick One

Consider a sequence of n operations performed on a data structure, where c_i and $\hat{c}_i$ denote the actual and the amortized costs of operation i, respectively. Which ONE of the following inequalities is essential for amortized complexity analysis?

(i) $c_i \le \hat{c}_i$

(ii) $c_i \ge \hat{c}_i$

(iii) $\sum_{i=1}^{n} c_i \le \sum_{i=1}^{n} \hat{c}_i$

(iv) $\sum_{i=1}^{n} c_i \ge \sum_{i=1}^{n} \hat{c}_i$

Assignment Project Exam Help
https://powcoder.com
Add WeChat powcoder

2. [10 points] SUPPORTING YOUR CLAIM
Pick any TWO of the statements in Question 1 (a)-(g) that you decided to be TRUE or FALSE. Give a complete proof of your decision.

Assignment Project Exam Help

https://powcoder.com

Add WeChat powcoder

Assignment Project Exam Help
https://powcoder.com
Add WeChat powcoder

3. [16 points] Amortized Analysis

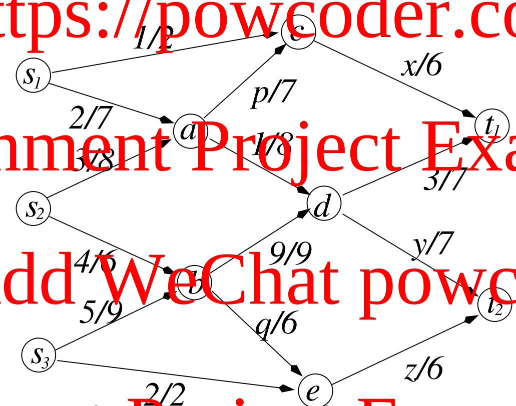
Recall the amortized analysis of a sequence of n insertions into a dynamic table whose capacity is doubled every time it becomes full. (As usual, inserting a new item into an empty slot takes unit time, as well as transfering each item into the expanded table.) Here, we consider two alternative expansion strategies whenever the table is full.

(a) Table size is tripled, i.e., it goes from T to $3T$. Use *Aggregate (or Potential) Method* to give an upper bound on the amortized cost of each insertion operation.

(b) Table size is increased by 1000, i.e., it goes from T to $T + 1000$. Use *Accounting Method* to give an upper bound on the amortized cost of each insertion operation.

4. [28 points] Flow Network with Multiple Sources/Sinks

Consider a variant of the Flow Network problem where we have multiple sources and multiple sinks. Figure shows a flow network with three sources s_1, s_2 and s_3 and two sinks t_1 and t_2 on which a flow has been assigned. The two numbers on each edge shows the flow and the capacity values, respectively.



(a) Describe a simple transformation that turns a flow network with multiple sources/sinks into a flow network with single source/sink, so that can we can use the same exact algorithms described in class.

(b) What are the values of p, q, x, y, and z that make the assigned flow feasible?

(c) What is the value of the total flow out of all sources? Is this a maximum flow in this network?

(d) Draw the residual graph for this flow.

(e) Find a minimum $\{s_1, s_2, s_3\} - \{t_1, t_2\}$ cut in this network. What is the capacity of this minimum cut?

(f) Starting with a **zero flow** consider a sequence of three augmentations: (i) $< s, s_1, a, d, t_2, t >$ with flow 7, (ii) $< s, s_2, a, c, t_1, t >$ with flow 6, and (iii) $< s, s_3, b, d, t_1, t >$ with flow 7. Give a fourth augmentation that can follow these. Is there a fifth possible?

(g) Start with a **zero flow** in this flow network, and illustrate that the number of augmentations performed by the Edmonds-Karp algorithm can be more than the number of augmentations performed by the Ford-Fulkerson algorithm. (Simply list two sequences of augmentations with their flow values, one for each algorithm.)

Assignment Project Exam Help
https://powcoder.com
Add WeChat powcoder

5. [20 points] MST Update

You are given an undirected connected graph $G = (V, E)$ with positive edge weights, and a minimum spanning tree $T = (V, E')$ of G with respect to those weights. You may assume G and T are given as adjacency lists. Now suppose the weight of a particular edge $e \in E$ is modified from $w(e)$ to $\hat{w}(e)$. You wish to quickly update the minimum spanning tree to reflect this change, without recomputing the entire tree from scratch. For each of the following cases, decide whether an update might be necessary, and if so describe a linear-time algorithm for updating the tree.

(a) $e \notin E'$ and $\hat{w}(e) > w(e)$

(b) $e \notin E'$ and $\hat{w}(e) < w(e)$

(c) $e \in E'$ and $\hat{w}(e) > w(e)$

(d) $e \in E'$ and $\hat{w}(e) < w(e)$

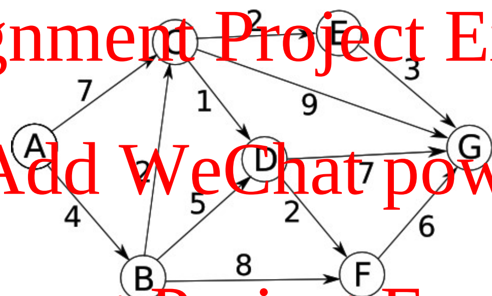
Assignment Project Exam Help
https://powcoder.com
Add WeChat powcoder

6. [14 points] Shortest Paths

For the directed weighted graph shown below, use Dijkstra's algorithm to compute the shortest paths from node A to all other nodes by filling in the table. At each step add a new vertex to M, the set of nodes whose shortest path length from A is correctly computed. The first two steps are already given.

$d(X)$: the cost of the current shortest path estimate from A to node X.

$p(X)$: the predecessor of node X along the current shortest path estimate from A.



Step	M		A	B	C	D	E	F	G
0	-	d	0	∞	∞	∞	∞	∞	∞
		p	-	-	-	-	-	-	-
1	A	d	0	4	7	∞	∞	∞	∞
		p	-	A	A	-	-	-	-
2		d							
		p							
3		d							
		p							
4		d							
		p							
5		d							
		p							
6		d							
		p							
7		d							
		p							

- Trace back on array p to output the shortest path from A to G:

Assignment Project Exam Help
https://powcoder.com
Add WeChat powcoder