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., $\gamma(p, q) + \gamma(q, r) \leq \gamma(p, r)$ for every p, q, and r in V.

   (b) TRUE/FALSE
   A *Bottleneck spanning tree* of an undirected graph G is defined as a spanning tree of G whose largest edge weight is minimum over all spanning trees of G. A bottleneck spanning tree of a graph is also a minimum spanning tree.

   (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 weights, 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
   If an edge e is part of some minimum spanning tree of an undirected, connected graph G, then it must be a lightest edge across some cut of G.

   (f) TRUE/FALSE
   Let $f_1, f_2 : V \times V \to R$ be two different flows on a flow network $G = (V, E)$ with a capacity $c : V \times V \to R^+ \cup \{0\}$. 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.

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

(g) True/False

If all edge capacities in a flow network are integer multiples of 9, then the maximum flow value is always a multiple of 9.

(h) 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 nondecreasing order of edge weights w
$T \leftarrow E$
for each edge e, taken in nondecreasing order by weight
    do if $T - \{e\}$ is a connected graph
        then $T \leftarrow T - e$
return T

(i) Pick One

Let $G = (V, E)$ be a directed graph with edge-weight function $w : E \to R$. 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) $d_{ij}^{(m)} = \min_{i \le j}\{d_{ik}^{(m-1)} + a_{kj}\}$ for $k = 1, 2, \ldots, n$
(iv) $d_{ij}^{(m)} = \min_{1 \le k \le n}\{d_{ik}^{(m-1)} + a_{kj}\}$ for $m = 1, 2, \ldots, n-1$

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

2. [10 points] SUPPORTING YOUR CLAIM
Pick any TWO of the statements in Question 1 (a)-(h) 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

3. [20 points] AMORTIZED ANALYSIS

A sequence of n operations is performed on a data structure. The kth operation has a cost of k when k is an exact power of 4, and otherwise it has a cost of 1. Analyze the amortized cost per operation using any one of these methods: (a) aggregate, (b) accounting, (c) potential. Specify which method you choose.

https://powcoder.com

Assignment Project Exam Help

Add WeChat powcoder

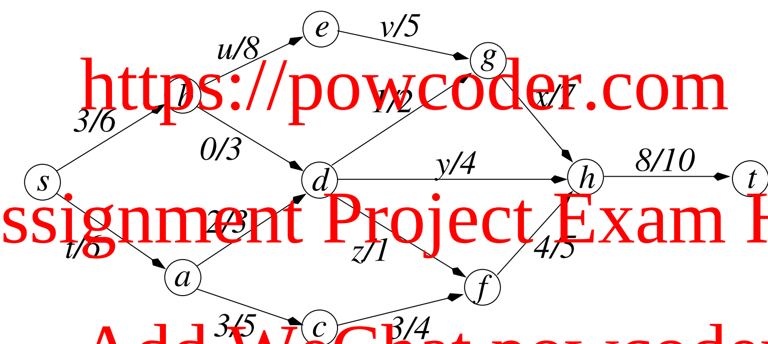
Assignment Project Exam Help

https://powcoder.com

Add WeChat powcoder

4. [25 points] Maximum Flow

Figure shows a flow network on which an $s-t$ flow has been assigned. The two numbers on each edge shows the flow and the capacity values, respectively.



(a) What are the values of t, u, v, x, y, and z that make the assigned flow feasible?

(b) What is the value of this flow? Is this a maximum $s-t$ flow in this network?

(c) Draw the residual graph for this flow.

(d) Find a minimum $s-t$ cut in this network. What is the capacity of this minimum cut?

(e) Starting with a **zero flow** consider a sequence of three augmentations: (i) $< s, a, c, f, h, t >$ with flow 4, (ii) $< s, a, d, f, h, t >$ with flow 1, and (iii) $< s, b, d, h, t >$ with flow 3. Give a forth augmentation that can follow these. Is there a fifth possible?

(f) 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. [15 points] MST OF AN ALMOST TREE GRAPH

A graph $G = (V, E)$ is called an *almost tree* if it is connected and has most $n + c$ edges where $n = |V|$ and c is small constant number. Design and analyze and algorithm for a given *almost tree* graph G with distinct costs on its edges, computes a minimum spanning tree of G in time $O(n)$.

Assignment Project Exam Help

https://powcoder.com

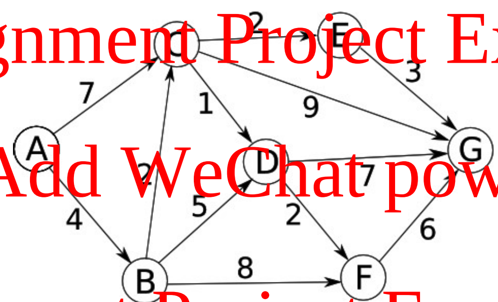
Add WeChat powcoder

6. [10 points] SHORTEST PATH ALGORITHM

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.



Assignment Project Exam Help

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

https://powcoder.com

Add WeChat powcoder

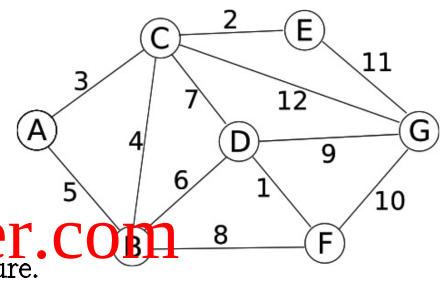
- Trace back on array p to output the shortest path from A to G:

https://powcoder.com

Assignment Project Exam Help

Add WeChat powcoder

7. [10 points] MINIMUM SPANNING TREES



Kruskal's algorithm for computing a Minimum Spanning Tree of an input graph G (as described in your textbook) considers the edges of G in non-decreasing order of weights and adds to a list of selected MST edges unless a cycle is formed by the selected edges, relying on the Union-Find data structure.

Union-Find maintains a collection of sets of vertices, those who are connected to each other with the selected MST edges. Union-Find is initially a collection of one vertex sets (see the first row of the table below). $\mathrm{find}(X)$ operation returns the ID of a set a vertex X belongs to. $\mathrm{union}(X, Y)$ operation merges two sets with a given member of each. Based on this description, complete the table below, showing iterations of Kruskal's algorithm on the given graph abow (first step is already given).

Step	Edge Considered	Sets in Union-Find Data Structure	Union-Find Operations	Select as MST Edge?
1	(D, F)	{A}, {B}, {C}, {D}, {E}, {F}, {G}	$\mathrm{find}(D) \neq \mathrm{find}(F)$ $\mathrm{union}(D, F)$	Yes
2				
3				
4				
5				
6				
7				
8				
9				
10				
11				
12				

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