Name: ______________________________ Section: ____________________

15-351 / 15-650 / 02-613 (Fall 2019): Final Exam (5:30pm–8:30pm, 12/9/2019)

Note: Please solve each of the following problems. This is a closed-notes and closed-book exam. You also should not use your laptops and cell phones. If you need additional space, use the back of the pages and indicate that.

Note: When you prove NP-completeness, you can assume that Vertex Cover, Set Cover, Independent Set, 3-SAT, Hamiltonian Cycle, Hamiltonian Path, which we discussed in class, are NP-complete.

Problem 1. *(18 points) Short answer.* (3 pts per question)

i. True or False: if the weights on a graph are distinct, the edge with the largest weight in the graph is never in the minimum spanning tree. Explain why.

ii. True or False: Let X be a decision problem. If we prove that X is in the class NP and give a polynomial-time reduction from X to 3-SAT, we may conclude that X is NP-complete. Explain why.

iii. State the maxflow-mincut theorem.

iv. True or False: a maximum flow in a network is unique. Explain why.

v. When we say X is a 3-approximation algorithm for problem Y, what exactly does 3 mean here?

vi. Suppose the smallest vertex cover in a graph $G = (V, E)$ is k. What is the size of the largest independent set in G?

Problem 2. *(16 points)*

(a) (8 pts) Let $G = (V, E)$ be a connected, undirected graph. Design an $O(|V| + |E|)$ time algorithm to compute a path in G that traverses each edge in E exactly once in *each direction*. Briefly explain correctness and provide runtime analysis.

(b) (8 pts) Suppose G is a directed acyclic graph with positive weight $d(u, v)$ on each edge. Let s be a vertex of G with no incoming edges and such that every other node is reachable from s through some path. Design an algorithm that is faster than Dijkstra's algorithm to compute the shortest paths from s to all other vertices in G. Briefly explain correctness and provide runtime analysis.

Problem 3. *(13 points)* We have a set of K shelves on which we want to place n books. The books have weights $w_1, w_2, \ldots, w_n$. We want the books to be in alphabetical order, so we cannot reorder them. So that we do not overload any shelf, we want to minimize the maximum total weight placed on any shelf.

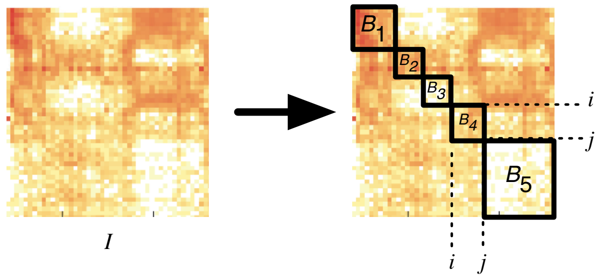
Example: Suppose $K = 3$ and the books' weights are:

w_1	w_2	w_3	w_4	w_5
10	6	10	2	15

then the lines above give a partition of the books into the shelves of value 17 (which is not optimal in this case).

Give a *dynamic programming recurrence* to compute the value of the partition of the weights into K groups that minimizes the weight of the heaviest group. Provide brief runtime analysis.

Problem 4. *(13 points)* You are given an image I that consists of $n \times n$ pixels of various colors. A *block diagonal partition* $\{B_1, \ldots, B_k\}$ of the image is a set of non-overlapping, *square* boxes that cover the diagonal of the matrix. See example below, where the boxes in the right image give a block diagonal partition.



You are also given a function called “$f(B)$” that returns a number between 0 and 1 to indicate how uniform the color within the box B is: it returns 1 if there is only one color in the box and 0 if there are lots of different colors.

Give a *dynamic programming recurrence* to find the optimal block diagonal partition $P = \{B_1, \ldots, B_k\}$ of an image I that maximizes the following quantity:

$$-\alpha k + \sum_{i=1}^{k} f(B_i)$$

Assignment Project Exam Help

where α is a given constant and k is the number of blocks in the partition P. Note you are given α, but you are *not* given k. Write the recurrence (including base case) with brief additional explanation for correctness. Provide brief runtime analysis.

https://powcoder.com

Add WeChat powcoder

Problem 5. *(15 points)* We consider HYPER-COMMUNITY as the problem that takes a collection of n web pages and an integer k, and determines if there are k web pages that all contain hyperlinks to each other. Prove that the HYPER-COMMUNITY problem is NP-complete.

Problem 6. *(15 points)* Given an undirected graph G with nodes $v_1, ..., v_n$ and $n \times n$ symmetric matrix R of natural numbers, and an integer b, is there a set S of b edges of G with the following property: Between nodes v_i and v_j, $i \neq j$, there are at least R_{ij} disjoint paths (that is, paths sharing no other node except for the endpoints) with edges in S. Prove that this problem is NP-complete.

Assignment Project Exam Help

https://powcoder.com

Add WeChat powcoder

Problem 7. *(10 points)* **(This question is optional** *only if* **you are in section 15-351)** You are studying the genetic cause of the disease *chronic serodiscipuli*, and you are working with a group of people $D = \{D1, D2, D3, \dots\}$ who have this terrible disease. For your study, you want to find a group of people C from a large population of people $P = \{P1, P2, P3, \dots\}$ who do not have the disease but have similar genetic mutations. More formally, you are given a table like the following:

		G1	G2	G3	G4	
People with the disease	**D1**	0	1	1	0	
	D2	0	1	1	1	
	D3	1	0	1	0	
	D4	1	1	0	0	
	D5	0	0	0	0	
	⋮					
People without the disease	**P1**	1	1	1	1	smoker
	P2	1	1	0	1	non-smoker
	P3	1	1	0	0	smoker
	P4	1	0	0	1	smoker
	P5	0	0	0	0	non-smoker
	P6	0	1	1	0	non-smoker
	⋮					

1 means person D2 has a mutation in gene G4; 0 means they don't.

where each row represents a person (either someone with the disease or without) and the entry in column G_i is 1 if the person a mutation in gene G_i, and is 0 if not. Additionally, you know whether each person in the non-disease group is a smoker or not.

Use *network flow* to design an algorithm to choose a set C of $k|D|$ people from population pool P such that:

(1) Every person in D shares at least m mutations with at least k people in your chosen set C, and

(2) No more than $k|D|/3$ of the people you choose are smokers.

(Note that k and m are parameters that you are given.)

Problem 8. *(12 points)* **(Extra Credit – Optional)** DOMINATING SET PROBLEM: Given a graph $G = (V, E)$ and an integer K, does G contain a dominating set of size at most K? A *dominating set* for a graph $G = (V, E)$ is a subset D of V such that every vertex not in D is adjacent to *at least* one member of D. Prove that the DOMINATING SET PROBLEM is NP-complete.