15-251: Great Theoretical Ideas In Computer Science

Fall 2013

Test Solutions 2

Problem	Points	Score	Problem	Points	Score
1	6		5	6	
2	6		6	20	
3	6		7	25	
4	6		8	25	
			Total	100	

Part I: Short answers ($5 \times 6 = 30$ points)

Explanations provided in multiple choice questions are only for your understanding. You do not need to give any justification for these questions in the exam, unless specified otherwise.

1. [6 points]
What is the relation between $f(n) = 2^{\log^* n}$ and $g(n) = n$? (Multiple answers may be correct.)

✓a) $f(n) \in O(g(n))$
b) $f(n) \in \Theta(g(n))$
c) $f(n) \in \Omega(g(n))$
d) None of the above

2. [6 points]
Consider the following problem: Given an undirected graph G and a number k, determine whether all *maximal* (not necessarily maximum cardinality) matchings of G are of size *at most* k. Which of the following statements are known to be true about this problem? (Assume $P \neq NP$. Multiple answers may be correct.) [Hint: A maximum cardinality matching is also a maximal matching.]

✓a) It is in P.
✓b) It is in NP.
c) It is NP-complete.
d) It is outside NP.

Why? Note that determining whether all maximal matchings are of size at most k is equivalent to determining whether the maximum cardinality matching is of size at most k. Since this matching itself can be computed in polynomial time, the problem is in P.

3. [6 points]
Recall the CleverVoting system from Homework 8 where each voter gives a fixed decreasing sequence of points ($\alpha_1 > \alpha_2 > \ldots > \alpha_k$) to candidates in his preference list in that order (instead of simply giving $k-1, k-2, \ldots, 0$ points as in the Borda count). Which of the following properties would every such voting rule (independent of the choice of αs) satisfy? (Multiple answers may be correct.)

✓a) Unanimity (if all voters put the same candidate first, it must be the winner)
b) Majority consistency (if a majority of the voters put the same candidate first, it must be the winner)
c) Condorcet consistency
d) None of the above

Why? Unanimity is clearly satisfied by all such rules since a unanimous candidate would clearly have a higher score than any other candidate. We saw in Homework 8 that no such rule satisfies Condorcet consistency. Note that Borda count does not satisfy Majority consistency (easy exercise!).

4. [6 points]
Consider the ski rental algorithm that rents for $3B$ days and then buys (buying costs $\$B$, renting costs $\$1$). Of the following four values of c, what is the smallest one such that this algorithm is c-competitive?

a) 2 b) 3 [✓c) 4] d) 5

Why? The worst-case for this algorithm is when it rains right after the algorithm buys, that is, if there are exactly $3B + 1$ sunny days. In this case, the optimal cost would be B (buying on the first day), while the algorithm pays a cost of $3B + B = 4B$. Hence, the approximation factor of the algorithm is 4 (and not any better than 4).

5. [6 points]
Consider the problem "CLIQUE" of finding the maximum size of any clique in a given graph, and the algorithm GREEDY that starts from the highest degree vertex and successively adds its neighbours as long as the added vertices remain a clique. Give an example of an undirected graph (possibly not connected) where GREEDY is NOT a $(1/2)$-approximation for CLIQUE.

ANSWER: Consider the graph consisting of two disconnected parts. One part is K_5, and the other part is a 6-star, i.e., a vertex v connected to 6 distinct vertices which have no edges between them. The GREEDY algorithm would start from v of the 6-star and would be able to add just one more vertex. The optimal solution is to pick the 5 vertices of K_5, showing that GREEDY is not a $(1/2)$-approximation.

Part II: (Variant of) Homework Question (20 points)

6. [20 points]
Let $G = (V, E)$ be a directed graph such that for every two vertices $u, w \in V$, exactly one of the two directed edges $u \to w$ and $w \to u$ is in E. Prove that G has a Hamiltonian path.

ANSWER: Such graphs are called "tournaments". Let $|V| = n$. We prove this using induction on n.

Base Case: For the base case $n = 1$, the vertex itself is a Hamiltonian path.

Induction Hypothesis: Suppose this holds for all tournaments with $n - 1$ vertices.

Induction Step: Consider a tournament G with n vertices. Take any vertex v and remove it from the graph. By induction hypothesis, there is a Hamiltonian path $v_1 \to v_2 \to \ldots \to v_{n-1}$ in the remaining tournament. Now we want to add v to this path to form a Hamiltonian path of G. If there is an edge $v \to v_1$, then we can add v in the beginning. If there is an edge $v_{n-1} \to v$, then we can add v to the end.

The only remaining case is when there are edges $v_1 \to v$ (incoming to v) and $v \to v_{n-1}$ (outgoing from v). Thus, look at the minimum i such that there is an edge $v \to v_i$. We must have $v_{i-1} \to v$. Thus, inserting v between v_{i-1} and v_i gives a Hamiltonian path for G.

Part III: Longer Problems ($2 \times 25 = 50$ points)

Remember that in this part, you need to formally prove your answer. Be crisp, precise, and accurate!

7. [25 points]
Suppose you have n indivisible objects (say a keyboard, a mouse, a monitor etc.) and two players. Each player i has some value $v_{ij} \in \mathbb{R}_{\geq 0}$ for each object j. Prove that checking whether there exists an envy-free division of these objects between the two players is NP-complete. Use PARTITION as your NP-complete problem. [Hint: Create identical valuations!]

ANSWER: First, we notice that the problem is in NP: given a division of objects between the two players, it is easy to check if the division is envy-free by comparing the value of each player for his own objects with his value for the other player's objects.

To prove NP-hardness, we reduce from PARTITION. Take an instance $e_1, \ldots, e_n$ of PARTITION where the question is to check if there exists a partition of $\{1, \ldots, n\}$ into S and T such that $\sum_{j \in S} e_j = \sum_{j \in T} e_j$. Consider an instance of indivisible envy-free division where there are n indivisible objects, and $v_{1j} = v_{2j} = e_j$ for $1 \leq j \leq n$. That is, both players have identical valuations.

Therefore, a division of these objects is envy-free if and only if the sum of values of the objects allocated to each player is no greater than the sum of values of the objects allocated to the other player, i.e., if and only if the sum of values of the objects allocated to both players are equal. However, this is true if and only if the answer to the PARTITION instance is YES. Therefore, this is a valid reduction. Note that this is clearly a polynomial-time reduction.

Hence, the problem of determining if there exists an envy-free division of indivisible objects between two players is NP-complete.

8. [25 points]
Consider the following SET-COVER problem.

SET-COVER: Given a set of elements $U = \{e_1, \ldots, e_n\}$ and subsets $S_1, \ldots, S_t \subseteq U$ such that $\bigcup_{i=1}^{t} S_i = U$, find the minimum number of subsets whose union is U. That is, find the minimum number of subsets that need to be picked to "cover all elements".

Notice that this is different from the MAX-COVER problem from Homework 8; while MAX-COVER requires covering the *maximum number* of elements with a *given number of subsets*, SET-COVER asks to pick the *minimum number* of subsets to *cover all elements*. Thus, SET-COVER is a minimization problem. Consider the following GREEDY algorithm (same as the greedy algorithm for MAX-COVER). First pick the subset that has the maximum number of elements. Then, iteratively pick the subset that has the maximum number of yet uncovered elements.

Now, consider the following example. We have a class of students where there are 2^{i-1} boys and 2^{i-1} girls of each age $1 \leq i \leq k$. For each i, let A_i be the set of people of age i (there are 2^i of them). Let B be the set of boys, and G be the set of girls, so $|B| = |G| = \sum_{i=1}^{k} 2^{i-1}$. Consider the instance of SET-COVER where U is the set of all students in the class, and there are $k+2$ subsets available, namely $A_1, \ldots, A_k$, B and G.

Use this example to show that if GREEDY is an $f(n)$-approximation to SET-COVER, then $f(n) \in \Omega(\log n)$.

ANSWER: First, the number of students in U is $n = 2 \cdot \sum_{i=1}^{k} 2^{i-1} = 2 \cdot (2^k - 1)$. Hence, $k = \Omega(\log n)$. We note that the optimal solution only picks the two sets B and G to cover all the students. Now, we analyze the GREEDY algorithm.

Note that $|B| = |G| = 2^k - 1$, and $|A_i| = 2^i$ for $1 \leq i \leq k$. We prove by induction on the number of iterations that in iteration i, the GREEDY algorithm picks A_{k+1-i}.

Base Case: For $i = 1$, note that A_k has the maximum number of students among all A_i's, B and G. Hence, the GREEDY algorithm starts by picking A_k.

Induction Hypothesis: Assume that the GREEDY algorithm picks $A_k, A_{k-1}, \ldots, A_{k+1-i}$ in first i iterations.

Induction Step: In iteration $i+1$, note that the number of uncovered students that B or G contribute is $\sum_{i=1}^{k-i} 2^{i-1} = 2^{k-i} - 1$. However, A_{k-i} contributes 2^{k-i} uncovered students. Other A_i's would only contribute less number of uncovered students. Hence, the GREEDY algorithm would pick A_{k-i}.

By induction, the GREEDY algorithm would select $A_1, \ldots, A_k$ and stop (since all the students are covered). Since the optimal algorithm picks only two subsets instead of k, the approximation ratio of the GREEDY algorithm on this example is $k/2 = \Omega(\log n)$. Hence, the worst-case approximation ratio of the greedy algorithm must be $\Omega(\log n)$ as well.