CS 5633 Analysis of Algorithms – Spring 10

4/20/10

10. Homework

Due **4/27/10** before class

1. **Hamiltonian Cycle (4 points)**
   Given that the Hamiltonian cycle problem for undirected graphs is NP-complete, show that the Hamiltonian cycle problem for directed graphs is also NP-complete.

2. **Subgraph isomorphism (5 points)**
   The subgraph isomorphism problems takes two graphs G_1 and G_2 as input and asks whether G_1 is isomorphic to a subgraph of G_2. See page 1082/1171 for the definitions of graph isomorphisms and subgraphs. Show that the subgraph isomorphism problem is NP-complete.
   Hint: Show that the problem is in NP, and then show that it is NP-hard. For the NP-hardness you need to pick an NP-hard problem (ideally one that involes a graph and a subgraph), and polynomially reduce it to the subgraph isomorphism problem.

3. $\Pi_1 \leq \Pi_2$ **(8 points)**
   Let Π_1 and Π_2 be decision problems and suppose Π_1 is polynomial time reducible to Π_2, so, $\Pi_1 \leq \Pi_2$. Answer and justify each of the questions below:

   (a) If Π_2 is NP-complete, does this imply that $\Pi_1 \in NP$?
   (b) If Π_1 is NP-complete, does this imply that $\Pi_2 \in NP$?
   (c) If $\Pi_2 \in P$ does this imply that $\Pi_1 \in P$?
   (d) If $\Pi_1 \in P$ does this imply that $\Pi_2 \in P$?
   (e) If Π_1 and Π_2 are NP-complete, is Π_2 polynomially reducible to Π_1?
   (f) If $\Pi_1 \in NP$ does this imply that Π_2 is NP-complete?
   (g) If $\Pi_2 \notin P$ does this imply that $\Pi_1 \notin P$?
   (h) If Π_1 is NP-complete and $\Pi_2 \in P$, what does this imply?

4. **Vertex Cover for trees (5 points)**
   Develop a linear-time greedy algorithm that finds an optimal vertex cover for a tree. Argue why the cover computed by your algorithm is indeed an **optimal** vertex cover.

5. **Vertex Cover and Clique (4 points)**
   The vertex-cover problem and the clique problem are complementary in the sense that an optimal vertex cover is the complement of a maximum-size clique in the complement graph. (So, the reductions we showed on the slides actually work in both directions.)
   Give a counter example which shows that the approximation algorithm for vertex cover does not imply that there is a polynomial-time approximation algorithm with a constant approximation ratio for the clique problem.

Flip over to back page ⟹

Related questions from previous PhD Exams

Just for your information. You **do not** need to solve them for homework credit.

1. This problem is concerned with NP-completeness.

   Consider the following two decision problems.

   VERTEX COVER.
   Instance: An undirected graph $G = (V, E)$, and a positive integer k.
   Decision Problem: Is there a vertex cover of size k? A *vertex cover* is a subset $V' \subseteq V$ such that if $(u, v) \in E$, then $u \in V'$ or $v \in V'$ (or both).

   INDEPENDENT SET.
   Instance: An undirected graph $G = (V, E)$ and a positive integer k.
   Decision Problem: Is there an independent set of size k? An *independent set* is a subset $V' \subseteq V$ such that each edge in E is incident on at most one vertex in V'.

   (a) Define *polynomial-time reducibility.*

   (b) Show that INDEPENDENT SET is polynomial-time reducible to VERTEX COVER.

   (c) Suppose problem P_1 is polynomial-time reducible to problem P_2 ($P_1 \leq_{\mathrm{P}} P_2$). If there is a polynomial algorithm for P_1, what can be implied about P_2? If there is a polynomial algorithm for P_1, what can be implied about P_1?

   (d) Consider the complexity classes P, NP, NP-complete, and NP-hard. If P $\neq$ NP, what would be the subset relationships among these four classes? If P = NP, what would be the subset relationships among these four classes?

   (e) Assume that VERTEX COVER is an NP-complete problem. Use VERTEX COVER to show that INDEPENDENT SET is NP-complete. Specify what steps need to be done, and provide the details of your solution.

2. This problem is concerned with NP-completeness. Let Π_1 and Π_2 be two decision problems for which is known that $\Pi_1 \leq_P \Pi_2$ (i.e., Π_1 is polynomially reducable to Π_2). Briefly state what can be inferred (if anything) in each of the following cases.

   (a) $\Pi_1 \in NP$

   (b) $\Pi_2 \in P$

   (c) Π_1 is NP-hard

   (d) Π_2 is NP-complete