CS 5633 Analysis of Algorithms – Spring 08

4/17/08

11. Homework

Due **4/29/08** before class

1. **Transitivity (3 points)**

   Show the transitivity property of the polynomial-time reduction " $\leq$" (fact 3 on slide 17):
   Let Π, Π', Π'' be three decision problems. If $\Pi \leq \Pi'$ and $\Pi' \leq \Pi''$ then $\Pi \leq \Pi''$.

2. **To be or not to be... ... in NP (5 points)**
   Which of the problems below are in NP and which are not? Justify your answers.

   (a) Given a directed graph $G = (V, E)$ with non-negative edge weights, as well as two vertices $s, t \in V$. Compute a shortest path from s to t in G.

   (b) Given an undirected graph G. Is G a tree?

   (c) Given an unsorted array A of n numbers. What is the third largest element in A?

   (d) Given a positive integer i. Is i not a prime number (i.e., is it the product of two integers greater than 1)?

   (e) Given two undirected graphs G_1, G_2, are they isomorphic? (Two graphs $G_1 = (V_1, E_1)$, $G_2 = (V_2, E_2)$ are called *isomorphic* if there exists a 1-to-1 map $f : V_1 \to V_2$ such that $(u, v) \in E_1$ iff $(f(u), f(v)) \in E_2$).

3. **Subgraph isomorphism (5 points)**
   Problem 34.5-1 on page 1017.
   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, and polynomially reduce it to the subgraph-isomorphism problem. It would make sense to use a problem that involves a graph and a subgraph.

4. **Hamiltonian Cycle (5 points)**
   A Hamiltonian cycle in a graph is a simple cycle that contains each vertex in the graph. "Simple" means that the cycle cannot have any repetition, so, a Hamiltonian Cycle contains each vertex in the graph *exactly* once. The "Hamiltonian cycle problem" is: Given a graph, does it contain a Hamiltonian cycle?

   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.

FLIP OVER TO BACK PAGE $\Longrightarrow$

5. $\Pi_1 \leq \Pi_2$ **(10 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:

- Does this imply that Π_2 is NP-complete?
- If $\Pi_2 \in P$ does this imply that $\Pi_1 \in P$?
- If $\Pi_1 \in P$ does this imply that $\Pi_2 \in P$?
- If Π_1 is NP-complete does this imply that Π_2 is NP-complete?
- If Π_2 is NP-complete does this imply that Π_1 is NP-complete?
- If Π_2 is polynomially reducible to Π_1, are Π_1 and Π_2 both NP-complete?
- If Π_1 and Π_2 are NP-complete, is Π_2 polynomially reducible to Π_1?
- If $\Pi_1 \in NP$ does this imply that Π_2 is NP-complete?
- If $\Pi_1 \notin NP$ does this imply that $\Pi_2 \notin NP$?
- If $\Pi_2 \notin P$ does this imply that $\Pi_1 \notin P$?

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 ($P_1 \leq_P P_2$). 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.