Algorithms
Computer Science 140 & Mathematics 168
Instructor: B. Thom
Fall 2004
Homework 14a
Due on December 2, 2004 (beginning of class).
Holy Smokes—the semester is almost over!

1. [**15 Points**] **HITTING SET!**

   First, the gratuitous story: The Computer Science Department at The Western Institute of Technology (T.W.I.T.) has 5 faculty members, Dr. I. "Theo" Rize, Dr. I. Dutu, Dr. Sam E. Heer, Dr. Mia Thue, and Dr. Juan More. Each professor serves on a number of committees. The Polynomial Time Algorithms Committee consists of Professors Rize, Dutu, and More. The NP-Completeness Committee consists of Professors Dutu, Heer, and Thue. Finally, the Stock-the-Coke-Machine Committee consists of Professors Thue and More (lucky them). The Dean would like to convene a meeting in which the number of computer scientists invited is minimized (she can't stand their bad CS jokes!) but at the same time, each committee must be represented by at least one of its members. For example, inviting Professors Thue and More would be a solution in this case.

   The Dean has formulated this problem as the following general problem (known to computer scientists as the Hitting Set Problem): We are given a set S (the set of professors) and a collection, C, of subsets of S (the set of committees). A hitting set for C is a subset $S' \subseteq S$ such that S' contains at least one element from each subset in C. The Hitting Set optimization problem is to find the smallest hitting set.

   (a) Formulate the Hitting Set decision problem corresponding to the optimization problem described above.

   (b) In this part of the problem, you will prove that the Hitting Set decision problem is NP-complete.

      i. The first part of proving NP-completeness is to show that Hitting Set is in the class NP. Briefly, in one sentence or two, explain why this is true.

      ii. Next, we need to give a reduction from some known NP-complete problem to the Hitting Set (HS) Problem. Describe a reduction from the Vertex Cover (VC) Problem.

      iii. Next, explain why this reduction can be performed in polynomial time.

      iv. Finally, prove that your reduction is faithful. This is the most important part of the problem, so be careful and precise. (Recall: a faithfulness argument for a decision problem always requires two parts.)

2. [**15 Points**] **More on ILP!**

   In class we proved that Integer Linear Programming (ILP) is NP-complete. Our reduction was from 3SAT. In this problem you will prove that ILP is NP-complete using another reduction: Vertex Cover (VC). You do not need to show that ILP is in NP (we already argued that in class). You need only:

(a) Give a reduction.

(b) Explain briefly why the reduction can be done in polynomial time.

(c) Briefly address why your reduction is faithful.

While this question isn't difficult, it is great for evaluating your subtle understanding of reductions. Write up your reduction clearly and don't skip steps when proving faithfulness (for aspects of your argument that are obvious, a single sentence should suffice). Students often loose points because they are sloppy, overlooking arguably petty but important details. For instance, use the standard form for ILP problems presented in class when writing up your solution. (You can assume this form automatically includes the constraint that all variable values are greater than or equal to zero if you wish, but be sure to indicate that this constraint is being used if it matters when arguing your reduction's faithfulness.)