ArsDigita University
Month 5: Algorithms - Professor Shai Simonson

Problem Set 6 – NP-Complete Reductions

1. Reductions

a. Show that the Hamiltonian Path problem reduces to the Hamiltonian Circuit Problem and vice versa. (Assume undirected graphs).
b. The Set Splitting problem gives a set of elements and a collection of subsets, and asks if there is a way to split the set into two parts, such that each one of the subsets contains at least one element in each part. Prove that Set Splitting is NP-Complete. (Hint: Try reducing from Not All Equal 3SAT).

2. More Reductions

a. The Independent Set problem is to find the maximum number of vertices in a graph that are mutually not connected. Prove that Vertex Cover reduces to the Independent Set problem. (Hint: Think about the vertices NOT in the vertex cover.)
b. Prove that Independent Set problem reduces to Clique.
c. Prove that Not-All-Equal-3SAT reduces to 4-Colorability.

3. The Clique Problem

a. Design a polynomial time algorithm for the Clique problem for fixed clique size four.
b. Analyze its complexity.
c. Explain how this implies that Clique is not NP-Complete for planar graphs. (Hint: Look up Kuratowski's Theorem in Rosen).

4. The Coloring Problem

Prove that if the maximum degree of a graph is two or less, then you can solve the coloring problem in polynomial time.

5. Partition Revisited

a. Is the Partition problem, when the sum of all the numbers is a perfect square, still NP-Complete, or is there a polynomial time algorithm to solve it? Justify your answer with an algorithm or a reduction.
b. The Subset Sum problem is defined as a decision problem below:
Input: Finite set A, positive integer size $s(a)$ for each for each a in A, positive integer B.
Question: Is there a subset $A' \subseteq A$ such that the sum of the sizes of the elements in A' is exactly B?

Prove that Partition reduces to Subset Sum and vice versa. Explain why this implies that Subset Sum is NP-Complete in general, but can be solved in pseudo polynomial time.

6. The ADU Seating Problem

Some students here would like to see people reassigned to different cubicles, for various positive academic and social purposes. For the purposes of simplicity, let's assume that we have *n* students, and each one has a list of people to whom they would not object being assigned nearby. Let's also assume that the arrangement of the cubicles is done in a circle, so that each student has exactly two neighbors.

a. Write the problem formally in Input/Question form as a decision problem.
b. Prove that the problem is NP-Complete, even if each list contains at most three students.
c. Describe an algorithm to solve the problem when each list has at most two students.

7. Optional (Challenging): **ADU Seating Revisited**

a. Consider the variation when students make a list of the people they do *not* want to be nearby. What variations of the problem can you solve efficiently, and which are NP-Complete?
b. Consider variations where the layout is not onto a circle, but instead onto a straight line, a tree, a grid etc. Analyze these new problems.

8. Optional (Challenging): Shortest Bounded Path

The shortest bounded path problem asks for a given graph and two vertices, what is the shortest path of absolute value between the two vertices. This can be written formally as a decision problem as follows:

Input: A directed graph with positive and negative edge weights, two vertices *x* and *y*, and a positive integer bound *M*.

Question: Is there a path from *x* to *y* such that the sum of the edge weights is between $-M$ and M inclusive?

a. Is this problem NP-Complete or is there a polynomial time algorithm? Justify your answer.
b. Same question for the restricted case when the graph is a directed acyclic graph.