CSC373— Algorithm Design, Analysis, and Complexity — Spring 2016

Tutorial Exercise 5: P, NP, and NP-Complete

---

1. **Set Packing.** The set packing decision problem is defined as follows:

   SetPack: Given a universe set U, a set of subsets $F = \{S_j \mid S_j \subseteq U,\ 1 \le j \le m\}$, and an integer k, does there exist $C \subseteq F$ with $|C| \ge k$ such that no two distinct elements $S_i, S_j \in C$ intersect (i.e., for all S_i, S_j in C with $S_i \ne S_j$ we have $S_i \cap S_j = \emptyset$)?

   (a) Denote the independent set decision problem by **IndepSet**. Show **IndepSet** $\le_p$ **SetPack**.

   (b) Define **searchSetPack** to be the search problem for set packing. That is, given U and F as in the Set-Packing decision problem, find a subset $C \subseteq F$ such that $|C|$ is the maximum possible and no two distinct elements in C intersect.

   Prove that **searchSetPack** $\le_p$ **SetPack**.

   Hint 1: You need to first find k^*, the maximum possible size $|C|$. Then find the elements of C.

   Hint 2: In the second stage of the algorithm, where you build up a solution C, it is useful to write a loop invariant stating that the current solution "is promising" (i.e., in the same sense that we used for proving the correctness of greedy algorithms).

2. **3D Matching.** We consider the following two types of 3D matching problems.

   partial3DM: Given three distinct sets X, Y, and Z, with $|X| = |Y| = |Z| = n$, a set of triples $T \subseteq X \times Y \times Z$, and an integer k, does there exist a subset of triples $C \subseteq T$ of size $|C| \ge k$ such that no two distinct elements $C_i, C_j \in C$ have any element in common (i.e., if $C_i = (C_{i,1}, C_{i,2}, C_{i,2})$ and $C_j = (C_{j,1}, C_{j,2}, C_{j,2})$ are distinct triples in C then, for each $p = 1, 2, 3$, we have $C_{i,p} \ne C_{j,p}$)?

   perfect3DM: The input is similar to **partial3DM** except no integer k is input for this problem. The question here is whether there exists a set of triples $C \subseteq T$ such that $|C| = n$ and no two distinct elements $C_i, C_j \in C$ have any of their three elements in common? (That is, the matching is perfect in the sense that each element of X, Y, or Z is covered by exactly one triple in C.)

   Note that Wikipedia defines the "3D matching problem" to be the problem **partial3DM** above, while the Kleinberg and Tardos text defines it to be **perfect3DM**. Moreover, in your answers below you can use the fact that **3-SAT** $\le_p$ **perfect3DM**, which is proved in the Kleinberg and Tardos text.

   (a) Show **perfect3DM** $\le_p$ **partial3DM**.

   (b) Show **partial3DM** $\le_p$ **SAT** by using an encoding of the constraints for 3D matching in terms of a CNF formula. Use the binary variables x_i, where x_i is true iff the i^{th} triple in T is to be included in the set C. (Note that we are asking simply for a reduction to **SAT**, not to **3-SAT**.)

   (c) Given the above results can you conclude that **partial3DM** or **perfect3DM** is NP-complete? Explain.

3. **Max Degree 12 Spanning Tree.** Show the following problem is NP-complete:

   Degree12Tree: Given an undirected graph $G = (V, E)$, does there exist a subgraph $T = (V, F)$ of G (i.e., with $F \subseteq E$) such that T is a spanning tree of G and the degree of every vertex in T is at most 12?

   Note, the **degree** of a vertex v in the graph (V, F) is defined to be the number of edges in F that have v as one of their endpoints.

   Hint: Consider making use of the Hamiltonian path decision problem.