**Spring 2016, CS 583: Approximation Algorithms**

**Homework 1**
Due: 02/17/2016

**Instructions and Policy:** Each student should write up their own solutions independently. You need to indicate the names of the people you discussed a problem with; ideally you should discuss with no more than two other people.

Read through all the problems and think about them and how they relate to what we covered in the lectures. Solve as many problems as you can. Please submit your solutions to at least 4 problems.

Please write clearly and concisely - clarity and brevity will be rewarded. Refer to known facts as necessary. Your job is to convince me that you know the solution, as quickly as possible.

**Problem 1** The Greedy algorithm for Max-$k$-Coverage and Set Cover can be implemented even if the sets in the set system are implicitly defined as long as we have the following oracle: given a subset $\mathcal{U}' \subseteq \mathcal{U}$ of the remaining uncovered elements from $\mathcal{U}$, return the set in the set system that covers the maximum number of elements from $\mathcal{U}'$. Some times we only have an *approximate* oracle. Suppose we only have an $\alpha$-approximate oracle for some $\alpha \leq 1$ which outputs a set from the set system that has at least $\alpha$ times the number of elements covered by the best set. Show that with an $\alpha$-approximate oracle Greedy gives a $(1 - e^{-\alpha})$-approximation. Note that this better than the easier bound of $\alpha(1 - 1/e)$.

**Problem 2** Consider a budgeted version of the maximum coverage problem. We are given $m$ sets $S_1, S_2, \ldots, S_m$, each a subset of a set $\mathcal{U}$. Each set $S_i$ has a non-negative cost $c_i$ and we are also given a budget $B$. The goal is to pick sets of total cost at most $B$ so as to maximize the number of elements covered.

- Show that if $c_i \leq \epsilon B$ for $1 \leq i \leq m$ then the Greedy algorithm yields a $1 - 1/e - \epsilon$ approximation (assume $\epsilon$ is sufficiently small if necessary).
- For any fixed $\epsilon > 0$ obtain a $1 - 1/e - \epsilon$ approximation. (Hint: consider the PTAS for knapsack from the lectures. You might find the inequality $1 - x \leq e^{-x}$ useful.)

**Problem 3** Problem 13.4 from Vazirani book.

**Problem 4** Problem 1.4 from Shmoys-Williamson book.

**Problem 5** Problem 3.6 from Williamson-Shmoys book.

**Problem 6** The multiple knapsack problem (MKP) is the following. Like in the standard knapsack problem the input consists of $n$ items, each of which has a profit $p_i$ and a size $s_i$. However, we are now given $m$ knapsacks with capacities $B_1, B_2, \ldots, B_k$.

- Describe a pseudo-polynomial time exact algorithm for the problem when $k = 2$.
- Prove that even for $k = 2$ and unit profits the problem is NP-Hard. Also prove that there is no FPTAS for the same setting. (Hint: use a reduction from the Partition problem.)

**Problem 7** Consider the MKP problem as in the previous problem. Consider a Greedy algorithm that picks each knapsack in turn and packs it using a $\alpha$-approximate algorithm for the single knapsack problem over the remaining items.

- Prove that if all knapsacks have the same capacity then you obtain a $(1 - e^{-\alpha})$-approximation by showing that the Greedy algorithm can be interpreted as solving a Max-$k$-Cover problem on an implicit set system.
- **Extra Credit:** Assume $\alpha = 1$. Show that the Greedy algorithm gives a 1/2-approximation even if the capacities are non-uniform and the order of the knapsacks is chosen arbitrarily.