CS 787: Advanced Algorithms	**Final Exam**
Out: 12/08/16	**Due: 12/13/16**

Guidelines

- This exam consists of 2 multi-part questions. You are required to solve and turn in both of the problems.
- Please typeset your solutions.
- You are not allowed to consult material outside of lecture notes and readings on the class webpage. You are not allowed to discuss the exam with your peers.

Problems

1. Suppose that you happen to come across a black-box for solving instances of the Knapsack problem exactly. You want to use this black-box for solving the following *Multiple Knapsack* problem: you are given n items of values v_i and weights w_i respectively, as well as k knapsacks of capacities $C_1, C_2, \cdots, C_k$. Your goal is to find a collection of disjoint subsets of items $S_1, \cdots, S_k$, such that each subset fits into a single knapsack, that is, for all $j \in [k]$, $\sum_{i \in S_j} w_i \leq C_j$, and the total value $\sum_{i \in \cup_j S_j} v_i$ is maximized.

   Consider the following greedy algorithm:

   - Use the single-Knapsack black-box to find the optimal subset of items for the first knapsack.
   - Recurse for the remaining knapsacks over the remaining items.

   (a) Prove that the greedy algorithm does not always return the optimal solution.

   (b) Prove that the greedy algorithm obtains a factor of 2 approximation. (Partial credit will be given for worse approximation factors, or for an analysis of a special case, e.g., $k = 2$ or $C_1 = C_2 = \cdots = C_k$.)

2. A fishmonger has one fish to sell every day. Every day one customer drawn randomly from a population U visits his stand. Customer $i \in U$ is willing to pay up to $\$v_i$ for the fish. If the fishmonger asks for a price of p for the fish with $p > v_i$, the customer walks away and the fish goes wasted. If $p \leq v_i$, then the fishmonger earns a revenue of $\$p$. The fishmonger's problem is to determine a good price to charge for the fish so as to maximize his expected daily revenue — the price times the probability that a customer drawn from U will buy the fish at that price. The problem is that the fishmonger does not know the values v_i for $i \in U$.

   For both of the following parts, you may assume that the values v_i lie in the range $[1, H]$. Let n denote $|U|$. Think of n as being very large, and H as being moderately large (so, in particular, $n \gg H$).

   (a) Suppose that the fishmonger is given the values v_i for a randomly drawn set $S \subset U$, with $|S| = m$. How should the fishmonger use this information to set a price? What revenue guarantee can you prove for your choice?

   (b) Suppose that the fishmonger wants to learn a good price to charge by observing customers over time. Here we will assume that customers are drawn adversarially rather than randomly from U. Every day the fishmonger announces a price, a customer arrives and decides whether or not to purchase the fish, and then, regardless of this decision reveals his value v_i. Devise a learning algorithm for the fishmonger that has small regret relative to the optimal single price to use in hindsight. Your algorithm should obtain regret that is sublinear in the number of days, T, but can depend on other parameters of the problem, such as H.

   Hint: The set of all possible prices is continuous; In order to apply a learning approach, you should first reduce the problem to optimizing over a discrete set of prices at some small loss in revenue.