

Homework I

*Due on 23rd August, 2026***Instructions:**

1. Submit as a pdf file (write your answers in Latex or Doc)
2. Do not copy from any source (including LLMs).

1. Give an example to show that the greedy algorithm for vertex cover can have $\Omega(\log n)$ approximation ratio, where n is the number of vertices in the graph.
2. Consider the following maximization version of the set cover problem. Given an instance consisting of a ground set U and subsets $S_1, \dots, S_m$ of U , and an integer k , the goal is to select k subsets such that the size of their union is maximized.
 - (i) Consider the following greedy algorithm. Repeat for k steps the following: select a set that covers the maximum number of uncovered elements. Show that this algorithm has $(1 - 1/e)$ -approximation ratio.
 - (ii) Consider the following variant of the above algorithm. In step i , let S_i^* be the set that covers the maximum number of uncovered elements. Instead of selecting S_i^* , the algorithm selects a set S_i that covers at least α fraction of the (new) elements covered by S_i^* (here α is a constant less than 1). Show that the approximation ratio of this algorithm is $(1 - 1/e^\alpha)$.
 - (iii) Write down a linear programming relaxation for this problem and show that the result in (i) above can also be derived by a randomized rounding algorithm.
3. Consider the makespan minimization problem. But now, each machine has a speed s_i and the time taken to process a job of size p_j on a machine i is p_j/s_i . Give a constant factor (greedy) approximation algorithm for minimizing the makespan.
4. Give a PTAS for the following problem: you are given a bag of size B , and n items. Each item i has a size s_i and a cost c_i . The goal is to select a minimum cost subset of items whose total size exceeds B . Write a linear programming relaxation for this problem. What can you say about its integrality gap?
5. Give an $O(\ln n)$ -approximation algorithm for the following problem: you are given a set U of size n and m subsets $S_1, \dots, S_m$ of U . Each set S_i has a cost c_i and each element $e \in U$ specifies a positive integer r_e . The goal is to select a sub-collection of subsets of minimum total cost such that for each element, at least r_e distinct sets containing it have been selected. You can assume that the input is such that there are at least r_e sets containing each element e .