- You have to discuss the running time of your algorithm. Always try to give algorithm with best possible running time. The points that you obtain will depend on the running time of your algorithm. For example, a student who gives an $O(n)$ algorithm will receive more points than a student who gives an $O(n^2)$ algorithm.
- You are required to give proofs of correctness whenever needed. For example, if you give a greedy algorithm for some problem, then you should also give a proof why this algorithm outputs optimal solution.
- **Use of unfair means will be severely penalized.**

There are 3 questions for a total of 50 points.

---

(10) 1. You are given an ordered sequence of n cities, and the distances between every pair of cities. You must partition the cities into two subsequences (not necessarily contiguous) such that person A visits all cities in the first subsequence (in order), person B visits all cities in the second subsequence (in order), and such that the sum of the total distances travelled by A and B is minimized. Assume that person A and person B start initially at the first city in their respective subsequences.

(20) 2. Ms. X wants to visit some shoe stores out of the n stores in the city $S_1, ..., S_n$. Mr. Y has to drive Ms. X around. He has to pick her up from her house and drop her back to her house. For each store S_i, there is a value $v(i)$ that denotes the satisfaction that Ms. X gets on visiting the store S_i. Mr. Y on the other hand, is concerned about the driving cost. For each pair of stores S_i, S_j there is an associated driving cost $d(i, j)$ that denotes the cost Mr. Y has to incur when driving between S_i and S_j. The driving cost from Ms. X's house to a store S_j is denoted by $d(0, j)$. You have to find a tour of a subset of stores starting and ending at Ms. X's house, that maximizes the total satisfaction of Ms. X minus the total driving cost incurred by Mr. Y.

(20) 3. You are given a rectangular piece of cloth with dimensions $X \times Y$, where X and Y are positive integers, and a list of n products that can be made using the cloth. For each product $i \in [1, n]$ you know that a rectangle of cloth of dimensions $a_i \times b_i$ is needed and that the final selling price of the product is c_i. Assume the a_i, b_i, and c_i are all positive integers. You have a machine that can cut any rectangular piece of cloth into two pieces either horizontally or vertically. Design an algorithm that determines the best return on the $X \times Y$ piece of cloth, that is, a strategy for cutting the cloth so that the products made from the resulting pieces give the maximum sum of selling prices. You are free to make as many copies of a given product as you wish, or none if desired.