You are allowed to discuss with others but are not allowed to use any references other than the course notes and the three reference books. Please list your collaborators for each question. You must write your own solutions. See the course outline for the homework policy.

The full mark is 50. This homework is counted 6% of the course grade.

---

1. **Programming Problem: Alice and Bob in Potatoland** (20 marks)

   Alice and Bob are farmers who own a vast piece of land. Since the ideal climate for potatoes to grow is approaching, Alice and Bob are looking for help in managing their potato plantation.

   Alice and Bob have their own set of potential *shifts*, where each shift has a starting time and a finishing time (both the starting and finishing times are included in the shift). These shifts need not be disjoint. Each farmer has three energy levels: they can be *exhausted* (energy level 0), *weary* (energy level 1) or *energetic* (energy level 2). A farmer's energy level decreases by at least 1 if they work a shift, and it increases by 1 if they do not work (i.e., are not scheduled for a shift) for k consecutive time units. This energy recovery can happen multiple times between two worked shifts. Energy levels are capped at at maximum of 2.

   Two values are associated with each shift: one corresponding to the number of potatoes produced if the farmer works the shift while energetic, and one corresponding to the yield if the farmer works the shift while weary. A farmer cannot work two overlapping shifts. If a shift is skipped, no potatoes are produced. If a farmer works a shift that does not overlap with the other farmer's worked shifts, their energy level decreases by 1. However, if they work a shift that overlaps with the other farmer's worked shifts, their energy level decreases by 2. A farmer cannot work a shift if doing so would cause their energy level to drop below zero.

   Help Alice and Bob maximize their farm's potato output!

   Input: The first line contains three numbers n_1, n_2 and k, representing the number of shifts of Alice, the number of shifts of Bob, and the time needed to regenerate one level of energy, respectively. Each of the next n_1 lines contains four space-separated integers that describe one of Alice's shifts: its starting time s_i^a, its finishing time f_i^a, its weary yield w_i^a and its energetic yield e_i^a. The final n_2 lines match the format of the previous n_1 lines and contain four space-separated integers s_i^b, f_i^b, w_i^b and e_i^b for Bob's shifts. It is guaranteed that the shifts of a given farmer are distinct (i.e., they differ pairwise in at least either the starting time or finishing time). There are no guarantees concerning the order of the shifts for either farmer.
   Input constraints: $1 \leq n_1, n_2, s_i^a, f_i^a, s_i^b, f_i^b \leq 1000$, $1 \leq w_i^a, e_i^a, w_i^b, e_i^b \leq 1000$, with $s_i^a < f_i^a$ and $s_i^b < f_i^b$ for all i. It is guaranteed that the maximum number of potatoes that can be produced fits in a signed integer (i.e., is less than $2^{31} - 1$). It is not guaranteed that the weary yield is smaller than the energetic yield for a given shift.

   Output: The first line contains the number of potatoes produced by the farm. The second line contains two space-separated integers n_1', n_2', representing the number of scheduled shifts for Alice and Bob respectively. The third line contains n_1' space-separated integers, representing the indices of Alice's scheduled shifts, and the fourth line contains n_2' space-separated integers, representing the indices of

Bob's scheduled shifts. The indices correspond to the positions of the shifts in the input.

Sample Input 1:
```
3 0 3
1 3 5 10
4 5 1 1
7 8 1 100
```

Sample Output 1:
```
110
2 0
0 2
```

Explanation: Bob has no scheduled shifts, so we only have to worry about Alice's. Alice is inclined to work the third shift with energy level 2. There is no harm in also working the first shift with energy level 2, which yields 10 potatoes. There are exactly 3 time units separating the first and the third shift, which is enough to replenish 1 energy level, allowing her to work the third shift at energy level 2.

2. **Written Problem: Sketching Data with Outliers** (10 marks)

We now examine how allowing outliers changes the data sketching problem from HW3. Recall that our input in this problem is a list of data points (x_i, y_i) with increasing x_i and an error tolerance ϵ. In the original problem, we were required to partition the x-coordinates into intervals from s_j to f_j and to compute a value h_j for each interval so that

$$\max_{i: s_j \le i \le f_j} |h_j - y_i| \le \epsilon.$$

In the variant we now consider, we can proclaim some subset of the data points to be outliers. We then remove the outliers from the data set, and solve the original problem on the remaining points. If $A \subset \{1, \ldots, n\}$ is the set of outliers, this is equivalent to requiring that

$$\max_{i: s_j \le i \le f_j \text{ and } i \notin A} |h_j - y_i| \le \epsilon. \tag{1}$$

The problem would be too easy if we were allowed to proclaim every point an outlier. Rather, we are told that we can label at most k points outliers. Our goal is to find a set A of size at most k and a partition of the input into intervals that satisfy (1) while using as few intervals as possible.

Design the fastest algorithm that you can to solve this problem. Prove the correctness and analyze the time complexity.

(Hint: You may begin by considering the following simpler variant of this problem: how can we find the smallest set A and value h so that

$$\max_{i: 1 \le i \le n, i \notin A} |h - y_i| \le \epsilon.$$

Once you can solve this, you should be able to solve the whole problem by dynamic programming.)

3. **Written Problem: Covering Set** (10 marks)

   Given an undirected graph $G = (V, E)$, a subset $S \subseteq V$ is a covering set if for every vertex $v \in V - S$ there exists $u \in S$ such that $uv \in E$ (i.e. every vertex $v \in V - S$ has a neighbor in S). We are interested in finding a covering set of minimum total weight, but this problem is hard on general graphs. Show that this problem is easy on trees.

   Input: A tree $T = (V, E)$ in the adjacency list representation, a weight w_v for each vertex $v \in V$.

   Output: A covering set $S \subseteq V$ that minimizes the total weight $\sum_{v \in S} w_v$.

   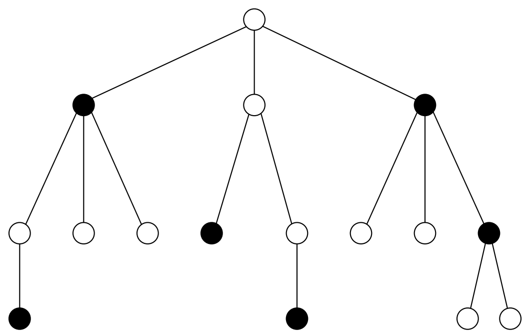

   Figure 1: The black vertices form a minimum covering set assuming all weights are one.

   Design the fastest algorithm that you can to solve this problem. Prove the correctness and analyze the time complexity.

4. **Written Problem: Colorful Path** (10 marks)

   Input: A graph $G = (V, E)$, where each vertex v has a color $c(v) \in \{1, \ldots, k\}$.

   Output: Find a path of k vertices in G such that all vertices on the path have distinct colors (i.e., each color $1, \ldots, k$ appears exactly once), or report that no such paths exist.

   Design an efficient algorithm for this problem. Prove the correctness and analyze the time complexity. You will get full marks if the proofs are correct and the time complexity is $O(2^k \cdot n^2)$.