1)Let S be a sequence of m integer pairs ⟨(x_1, y_1), (x_2, y_2), . . . , (x_m, y_m)⟩. Each of the values x_i and y_i , for all $1 \le i \le m$, is an integer between 1 and n. A domino sequence is a subsequence ⟨(x_{i1}, y_{i1}), (x_{i2}, y_{i2}),. . . , (x_{it}, y_{it})⟩ such that $1 \le i_1 < i_2 < \cdots < i_t \le n$ and, for all $1 \le j < t$, $y_{ij} = x_{ij+1}$. Note that it isn't necessarily true that $i_{j+1} = i_j + 1$, that is, the elements of the domino sequence don't have to be consecutive in S, but they have to appear in the right order.

Example: For S = ⟨(1, 3), (4, 2), (3, 5), (2, 3), (3, 8)⟩ , both ⟨(1, 3), (3, 5)⟩ and ⟨(4, 2), (2, 3), (3, 8)⟩ are domino sequences.

Use dynamic programming to find a longest domino sequence of S in O(n + m) time. Argue briefly that the running time of your algorithm is indeed O(n + m) and that its output is indeed a longest domino sequence of S.

2)You are given a set $X = \{x_1, x_2, . . . , x_n\}$ of points on the real line. Your task is to design a greedy algorithm that finds a smallest set of intervals, each of length 2, that contains all the given points.

Example: Suppose that X = {1.5, 2.0, 2.1, 5.7, 8.8, 9.1, 10.2}. Then the three intervals [1.5, 3.5], [4, 6], and [8.7, 10.7] are length-2 intervals such that every $x \in X$ is contained in one of the intervals. Note that 3 is the minimum possible number of intervals because points 1.5, 5.7, and 8.8 are far enough from each other that they have to be covered by 3 distinct intervals. Also, note that my solution is not unique – for example, I can shift the middle interval [4, 6] to the right, say to [5.7, 7.7], without disturbing the other intervals, and we would still have an optimal solution.

(a) Suppose that elements of X are presented in increasing order. Write a greedy algorithm code, running in O(n) time, for this problem.