Qn – 4: Design a greedy algorithm for covering all people with the minimum number of umbrellas. The input consists of the integers x1, x2, . . . xn, and L. The output of your algorithm should be the positions of umbrellas.

Answer:

Place the umbrellas at the farthest possible position to a person while it still covers the person

Algorithm:

Let y1, y2, … yn denote if the person is covered by the umbrella or not.

Let l1, l2, … lm denote the position of m umbrellas

Let last_umbrella_position be the position of the last umbrella placed. It is initialized to null at the start

Step 1: Sort x1, x2, … xn in ascending order of their positions

Step 2: For each person i … n:

If last_umbrella_position == null or xi > last_umbrella_position + L/2 :

Place umbrella j farthest to xi which still covers xi

i.e. lj = xi + L/2

Add lj to list of umbrella positions

last_umbrella_position = lj

else:

continue

Proof of correctness:

When each person is iterated, a new umbrella is added only if the person is not already covered by an umbrella

Consider an alternate optimal solution S which matches with the current algorithm S' at 1…j umbrella positions and differs in the j + 1 position.

Since, in S' the umbrellas are placed only if a person without cover is found and is placed farthest acceptable position. Therefore, lj+1 is S' is the farthest possible position j+1th umbrella can be placed

(i.e.) lj+1 in S' > lj+1 in S

If you replace the umbrella in lj+1 position in S with lj+1 from S', the number of umbrellas does not increase and it also covers maximum area still covering the required people

Proof of termination:

Algorithm iteratively runs for n number of persons and doesn't have any loops within or traces back to a person already visited. So, it terminates after n iterations.

Complexity:

Sort the position of the people – O(n log n) time complexity

Placing the umbrellas – Since it runs iteratively for n people and either places or does not place an umbrella – It takes only linear time O(n)

Total Time Complexity – O(n log n) + O(n) = O(n log n)

Space complexity – O(n)