CS5800: Algorithms Spring 2018
Assignment 4.2

Saptaparna Das

February 23, 2018

(a) Original problem
Given a set of random points $x_1, x_2, ..., x_n$ on a straight line and m being length of an interval, we have to calculate minimum number of intervals containing all the points.

(b) Algorithm:

```
1. Sort the distances  x_i  in nondecreasing order. Let a_0, a_1, a_2, a_3..., a_n be the sorted sequence
2. if n=0, return empty array
3. Set i to 0 and a_0 to result array
4. Repeat following steps, until i !=n :
        (a) Set current to a_i and distance to 0.
        (b) Increament i and do distance = a_i-current, until i != n and distance <= m
        (c) add a_i to result array, if distance > m
5. return result
```

(c) Proof of correctness:

To justify correctness of above greedy algorithm, we need to prove its optimal substructure property and greedy choice is the optimal choice.
1. Optimal substructure: Let S_{ij} be the distances of minimum number of light panels to hang on the wall to light up all the paintings. Let the distance x_k is included in the optimal solution. By including x_k in the optimal solution, we are left with two subproblems: finding minimum number of lights to cover distances from x_i to x_k and from x_k to x_j.
Let S_{ik} and S_{kj} be the set of distances from S_{ij} to cover x_i to x_k and from x_k to x_j. So, we can write $S_{ij} = S_{ik} + S_{kj} + x_k$ or $|S_{ij}| = |S_{ik}| + |S_{kj}| + 1$
Now, S_{ij} should include optimal solution of S_{ik} and S_{kj}.Because if there exists a set S'_{ik} which contains less number of light panels to cover x_i to x_k, then we can use S'_{ik} in the solution instead of S_{ik} . So, we can say, $|S'_{ik}|+|S_{kj}|+1 < |S_{ik}|+|S_{kj}|+1$, which implies $S'_{ij} < S_{ij}$ which is a contradiction, since S_{ij} is an optimal solution.

2. Greedy choice: To make greedy choice in this problem means to cover as much distance possible with single light panel. Since the distances are in non-decreasing order, starting from first painting, if we traverse distance of light panel and check for next avaliable painting after that, then we are left with only one subproblem to find optimal positions of light panels from that new light till end. We don't need to consider any other painting between first light and newly found light, since all such paintings will fall in horizontal span of first light panel. Now, to prove, the greedy choice is always part of some optimal solution,
consider S_k be a non-empty subproblem and x_m be the maximum distance between ist painting and one other painting in that range, which is less than or equal to length of light panel. Now, we have to show x_m is included in optimal solution for S_k.
Let A_k be the set of optimal distances in S_k and x_j is the maximum distance (less than or equal to length of light panel) we can cover from first painting, so that all the painings in that span are lit. Now, if x_m and x_j are not equal, then we can take x_j out of the set A_k and replace it by x_m to make new set A'_k. Since x_m is the maximum a light panel can cover from first painting, then x_m must be greater than or equal to x_j. In that case, $A'_k < A_k$, which is a contradiction to original assumption. So, x_m must be same as x_j

Efficiency:

Time taken to sort the distances is $O(n \log n)$. To compare distances and adding distances to result array happens linearly i.e. O(n) time. So time complexity should be $O(n \log n)$.
Space complexity is O(n) to store sorted elements. Apart from sorting the algorithm takes constant auxiliary space.