Assignment 1 CS345

Krishna Kumar Meena
160342

August 17, 2019

1. **Answer to Question 1:**

2. **Answer to Question 2:**

   Given n numbers which we want to sort with the help of convex hull. now imagine n numbers are x-coordinates and relation with y is-
   $$y = x^2 \tag{1}$$
   So we find an parabola of 2-D points and want convex hull of them.
   if we make a convex hull then we will find that all points are covered with convex hull. and convex hull is sorted according to x-coordinate.

   so we can sort number in O(nlogn) time using convex hull.

3. **Answer to Question 3:**
   A.
   Given Set of Points s $\subset R^2$,
   Definition of convex hull :- the convex hull of the set is the smallest convex polygon that contains all the points of it.

   So we can make many convex polygon that contain all the points inside it or on the polygon, to make smallest convex polygon we need to reduce points on the polygon. Which makes more point inside polygon and less points on it . Actually this process will decrease vertices of polygon with following property of convexing (All point of set S $\subset$R^2 should be inside the polygon or on it). At the end of this this process we will find an unique polygon which will be convex hull polygon.

   B.
   When we are making a convex hull we from a polygon we are making polygon with the points. then vectices are making from the points of S $\subset$R^2 . so all the vertices will be in S.

4. **Answer to Question 4:**

Given a set of Point Q $\subset$R^2 As input.

1. If Q = ø return; if | Q | = 1 return Q.
2. Split the input into left and right halves Ql and Qr, by computing the median xcoordinate among points of Q.
3. Let q have the maximum y-coordinate in Qr, and add q to the output set S.
4. Delete q and everything it dominates.
5. Recurse on (whats left of) Q and Qr.
We claim that This algorithm runs in O(n log h) time on every input.

Every recursive call successfully identifies a new maximal point (the point q). Thus the ith level of recursion identifies 2i new maximal points. Thus there can only be log k recursion levels, and the total work at each level is O(n). So time taken will be O(nlogh).

5. **Answer to question 5:**

Given two set A and B of integer of size n and that containing integer of size 0-10n.

Define a set C as catesian-sum multiset of set A + B

$$C = \left\{a + b \; |a \in A \text{ and } b \in B\right\}$$

So now in set C integer will be in range of 0-20n and can have maximum element n^2.

Now assume that an equation x- where a_i denote the occurrence of integer i in equation X. so we can relate it with set A.

$$\text{X}= \sum_{i=0}^{10n} a_i x^i$$

And same assumption for set B. where b_i denote the occurrence of integer i in equation Y. so we can relate it with set B.

$$\text{Y}= \sum_{i=0}^{10n} b_i x^i$$

Now if we multiply these two polynomial X and y then we will find another polynomial Z of maximum size of n^2. where z_i is showing occurence of integer i in Z. so Z will same as cartesian sum of two set

$$\text{Z}= \sum_{i=0}^{20n} z_i x^i$$

Now i have to find an algorithm to multiply two polynomial of size 10n in O(nlogn) time.

Fast-fourier transform (FFT) provides us a way to multiply two polynomials of degree-bound n in O(n lg n) time . so we will multiply these two polynomial and find other polynomial which can be convert to cartesian sum of of two set.