Problem Sheet - 5 - Solutions.

1) Let N be the set of nuts and B be the set of bolts. $|N| = |B| = n$ and for each $n \in N$, $\exists$ exactly one bolt b in B such that n fits perfectly in b.

~~IdNP & tP~~

~~N & B contains.~~

Algorithm:

1. ~~Pick~~ let current nut & bolt sets be (N, B).
2. pick $n' \in N$.
3. $\forall\, b \in B$ check if n fits into b.
4. Let $b' \in B$ fit with n perfectly
5. ~~Divide~~ Let B^L be the set of all bolts too small for n'.
6. Let B^R be the set of all bolts too large for n'.
7. Let N^L be the set of nuts too small for b'
8. Let N^R be the set of nuts too large for b'
9. Repeat from 1 on sets (N^L, B^L) & (N^R, B^R).

Time complexity.

Steps 3, 4, 5, 6, 7, 8 take $O(|N| + |B|)$ comparisons which is $O(n)$.

$\therefore$ The recurrence relation for expected run time.

$$T(n) = 2T\left(\frac{n}{2}\right) + O(n) = O(n \log n)$$

2) Given ~~Set~~ Set $A = \{(a_i, w_i)\}$ where each $a_i, w_i \in \mathbb{N}$.
We want to find a_j st.

$$\sum_{\substack{i \\ a_i < a_j}} w_i < \frac{1}{2} \sum_{i \in \mathbb{N}} w_i$$

&

$$\sum_{\substack{i \\ a_i > a_j}} w_i < \frac{1}{2} \sum_{i \in \mathbb{N}} w_i$$

A simple algorithm would be to sort the a_i and scan from left to right to check if such for such an a_j. This would take $O(n\log n) + O(n)$ time.

Consider the following alternative:

FindWeightedMedian(A, W) # A → array of a_i's / ~~w~~ weight
W → total weight of a_i's in A.

1) Find median ~~aqeda~~ $a' \in A$.
2) Let A^L be the ~~set~~ array containing all $a_i < a'$
2) Let A^R be the array containing all $a_i > a'$.
3) Let W^R, W^L, w' be the weights of A^L, & A^R and a' (resp)
4) If $W^L < \frac{1}{2}(W^L + W^R + w')$ & $W^R < \frac{1}{2}(W^L + W^R + w')$ return a'
5) If $W^L > \frac{1}{2}(W^L + W^R + w')$ return FindWeightedMedian(A^L, W).
6 If $W^R > \frac{1}{2}(W^L + W^R + w')$ return FindWeightedMedian~~(A', $W - W_L - w'$)~~
(A', $W - W_L - w'$)

It takes $O(n)$ time to find median & partition the array, and each time we discard half of the input.

$$\therefore \quad T(n) = T(n/2) + O(n) = O(n)$$

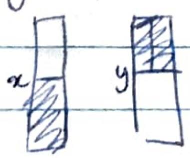
3) There are two databases DB_1 & DB_2 which contains n numbers each and all numbers are distinct. We want to ~~get~~ find the median of the $2n$ values by querying the data DB_1 & DB_2 for the k^{th} smallest value.

We can assume that DB1 & DB2 are sorted, since ~~also query~~ rank of each elem in DB1 & DB2 are available, however this may not be true.

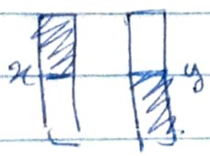
Suppose we query $n/2^{th}$ rank for DB1 & $n/2^{th}$ rank of DB_2.

let $x = DB1(n/2)$ & $y = DB2(n/2)$.

if $x < y$: then the rank of the overall median in DB1 is higher than $n/2$ & ~~the ran~~ & in DB2 it is lower than $n/2$.



if $y < x$: then the rank of the overall median in DB1 is lower than $n/2$ & in DB_2 it is higher than $n/2$



In both cases, we can discard half the numbers & recurse.
(unshaded region)

$\therefore$

Time complexity

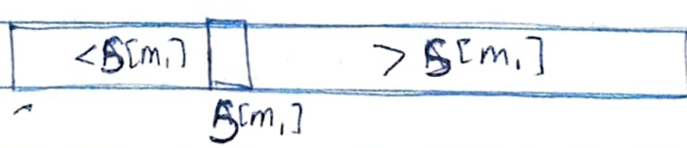
$$T(n) = T\left(\frac{n}{2}\right) + O(1) \rightarrow \text{checking if } x < y.$$

$$= O(\log n).$$

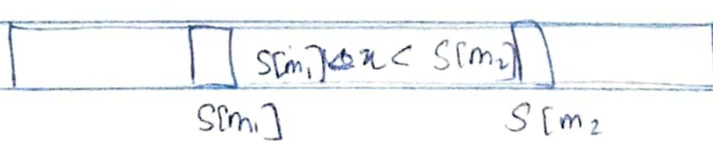
4) Given a set S, $|S| = n$. & two numbers $1 \leq m_1, m_2 \leq n$; we want to find all elements in S whose rank falls in the interval $[m_1, m_2]$ in $O(n)$ time.

→ Using standard select algorithm, we get the m_1^{th} & m_2^{th} element from S in $O(n)$

→ Next we partition array wrt $S[m_1]$



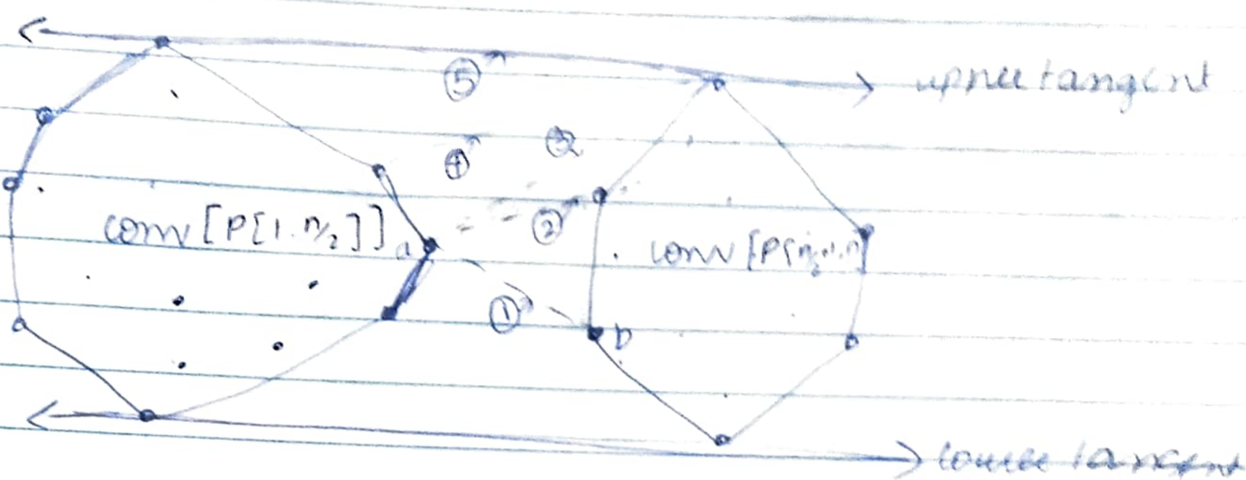
& then partition the right part wrt $S[m_2]$



Convex hull of a set of points P is the smallest convex set that contains all the points

Date:

5 Sort the points w.r.t x-coordinate. We recursively build the convex hull of the left half of the points & the right half.



We need to find the upper & lower tangents to merge the two convex hulls

Alg for upper tangent:

1) Let L be the line joining the rightmost point of conv[P[1..n/2]](a) & leftmost point of conv[P[n/2+1..n]](b).

2) While L crosses (intersects interior) of any of the polygons:

{while L crosses conv[P[1..n/2]]

$L \leftarrow L'$ → line joining a & b° = point on convex hull clockwise to b

while L crosses conv(P[n/2+1..n]

$L \leftarrow L'$ → line joining a: point on convex hull counterclockwise to a

&

Page No

Similarly, lower tangent is found.

lower & upper tangents merge the two convex hulls.

∴ The run time

$$T(n) = 2T(n/2) + O(n)$$ time to find tangents

$$= O(n \log n).$$