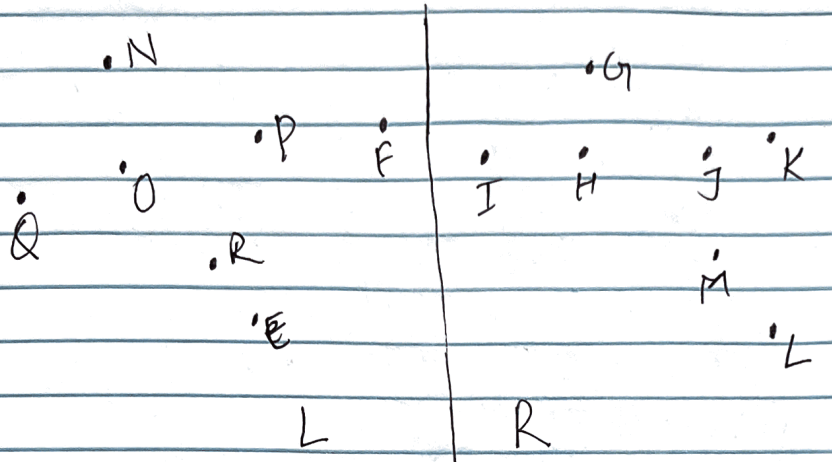
Design Experience 2 By: Pawan Acharya

Convex Hall using divide and Conquer

Problem:

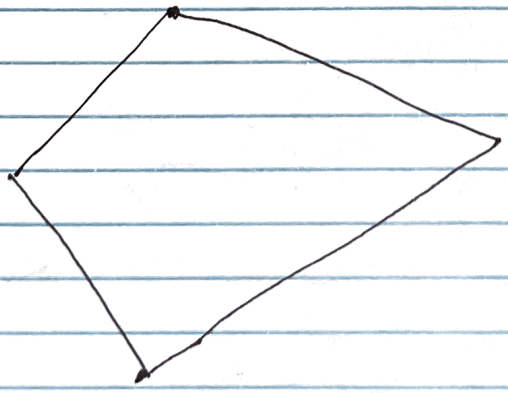
a set of 14 points (say), Trying to make a convex hall using divide and conquer

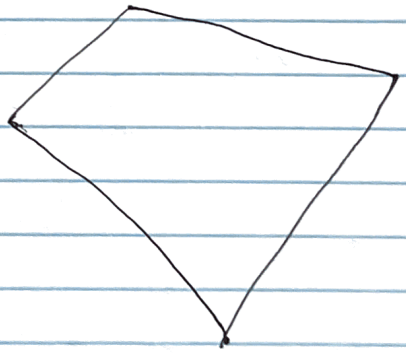


$n/2$ points = 7 points each subset.

① So, the convex hall of the subsets L and R are computed recursively.

Lets first sort them Left to Right. [Q, N, O, R, P E, F, I, H G, J, M, K, L]

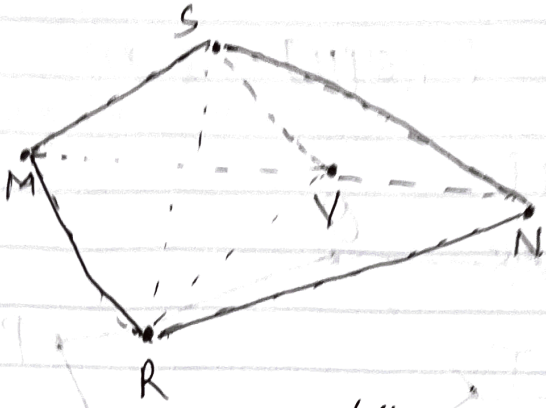




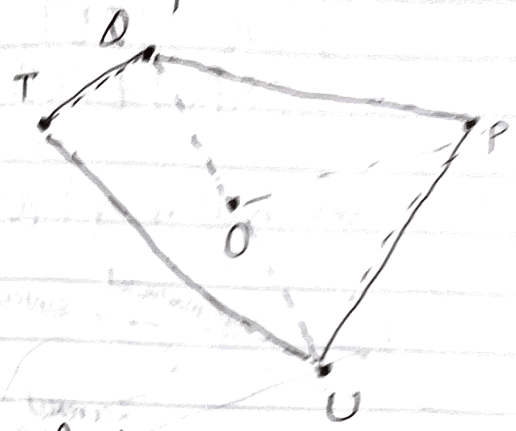
Design Experience 2

Pawan Acharya

Convex hall using divide and conquer approach (algorithm)
First of all let's say that we have 10 points.



Left



Right

First, let's divide all points into two subsets L and R. So, we have $n/2$ points in each subsets. In this case, each subset has 5 points as shown above. [The points sorted L to R : [M, R, S, V, N, T, O, Q, U, P]

Now, the conver hall of the subsets L and R are computed recursively.

To show this recursive approach in this case (mathematically) :

We have, Left subset : [M, R, S, V, N] = sorted L to R.

Each point is a convex hall i.e. has upper and lower tangent

Starting from left most point, convex hall for M is [M], and ri

[R]

so, [M] [R] [V] [N]

UL LL UR, LR

[V, N] [S]

[R, M] [S, N]

Left convex Hall ⇒ [R, M, S N] — clockwise

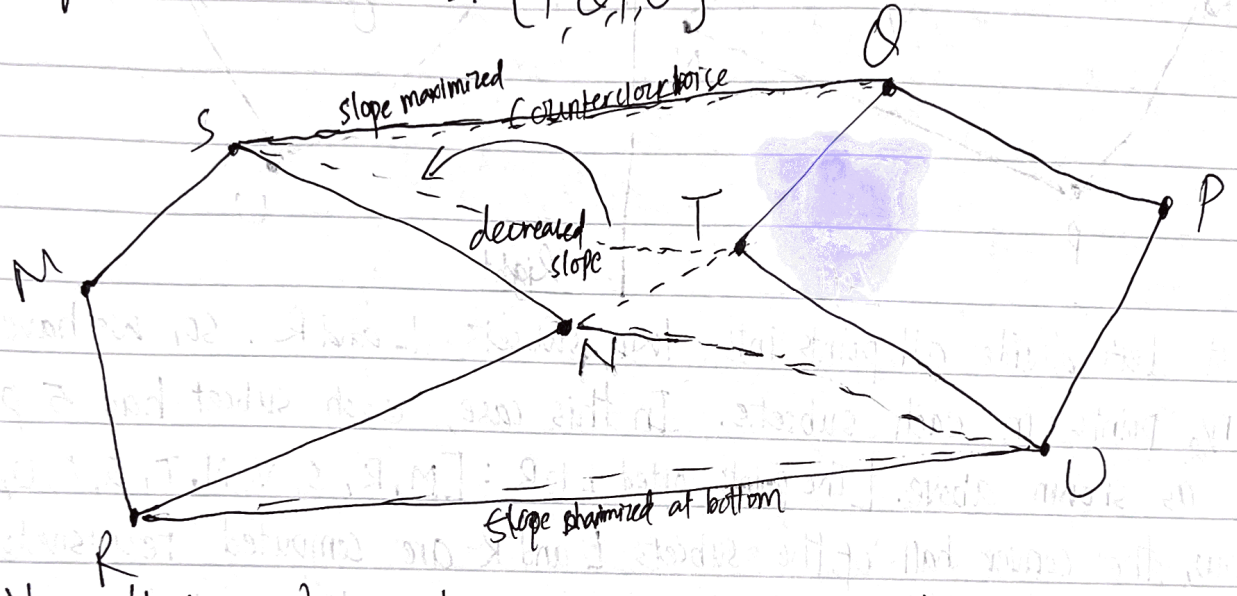
Similar for the right subset.

[T, Q] [Q, P]

[T Q P] [Q P]

~~[T Q P U] [T U]~~ [T, Q, P, U]

Right convex hall is: [T, Q, P, U]



Now that we have two convex halls, the next step is to merge them using tangent (Upper tangent ~~and~~)

~~let~~ Right most point of left polygon : N

Left most " " Right " : T

Counter clockwise order checking (from N to R until we find Upper tangent)

while (NT) (ST) (SQ)

while ✓

while

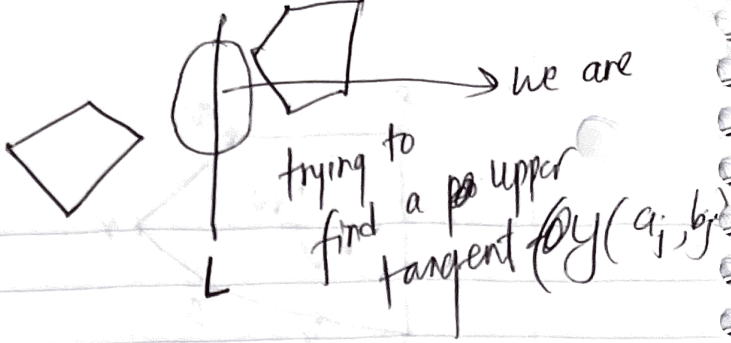
↓ slope maximized (doesn't cross the left polygon)

SO, SQ is Upper tangent ⇒

~~doing~~ Now for the bottom part; Lower right, lower left

NU RU

So, we got SQ and RU
And the overall convex hall is shown above.



Pseudocode for merge.

```
i = 1
j = 1
while (y(i, j+1) > y(i, j) or y(i+1, j) > y(i, j)):
    if (y(i, j+1) > y(i, j)):
        j = j+1 mod q        move right finger
    else
        i = i-1 mod p        move left finger
return (a_i, b_j) as upper tangent
```

of this

Complexity = $\Theta(n)$

merge [combine]

$$T(n) = 2T(n/2) + \Theta(n)$$

Overall convex hall $\Theta(n \log n)$

Arrangement of the overall hall.

cut and paste! First link (a_i, b_j), go down the b list till you see b_4, then link to a_k and continue till you return to a_i.

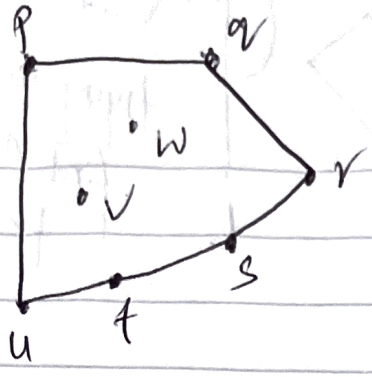
eg $a_2, b_1, b_1, b_2, b_3, b_4, a_1, a_2$.

$\Rightarrow$ order(n)

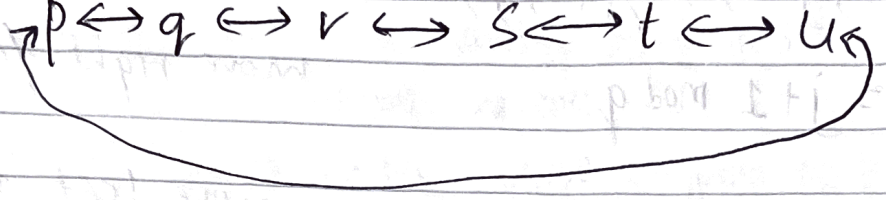
Brute force ? (without using the formula)

draw lines
Check if that includes all points in one side

For this we have n points
$O(n^2)$ → for making segments
$O(n)$ → for checking
Overall $(O(n^3))$ complexity in Brute force.



Sequence of points on the boundary of hull in clockwise order as doubly linked list.



Sort the points by x coordinate (once for all)
For input set S
Divide into left half A and right half B by x coordinates
Compute CH(A) and CH(B) recursively
Then combine (merge)

Big question: How to merge ?

two fingers algorithm.
max tangent.