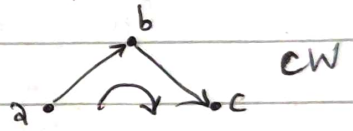
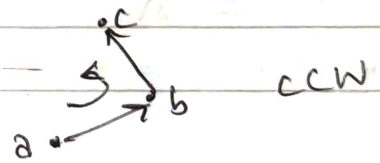
* Orientation

Three kinds of orientation for three pts. :

- Clockwise (CW) : right turn



- Counter CW : left turn



- Collinear (COLL) : no turn

Orientat^{n} can be characterized by sign of $\Delta(a,b,c)$

If $\Delta(a,b,c) < 0 \Rightarrow$ CW
$\Delta(a,b,c) = 0 \Rightarrow$ COLL
$\Delta(a,b,c) > 0 \Rightarrow$ CCW

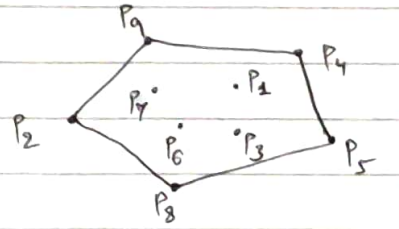
$$\Delta(a,b,c) = \begin{vmatrix} x_a & x_b & x_c \\ y_a & y_b & y_c \\ 1 & 1 & 1 \end{vmatrix}$$

* Convex Hull (CH)

- CH of a pt. set P, CH(P):
  → smallest convex set containing P
  → intersect^{n} of all convex sets containing P

- Give an algo. that computes convex hull of any given set of n pts. in the plane efficiently.
  i/p : locat^{n} of n pts.
  o/p : a convex polygon ⇒ a sorted sequence of pts. CW or CCW along the boundary.

- i/p = $P_1, P_2, P_3, \ldots\ldots, P_9$



o/p = P_2, P_9, P_4, P_5, P_8

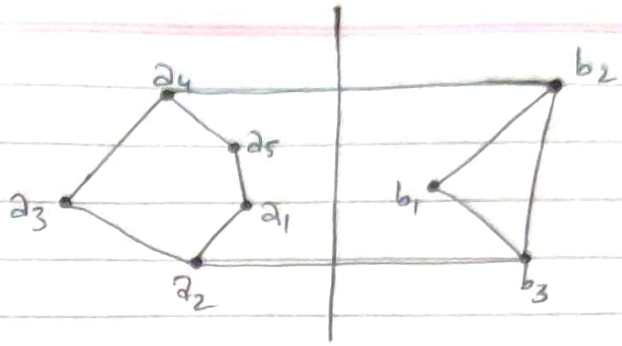
* Graham's Scan Algo.

Step 1: Find lowest pt p (guaranteed to be in)
Step 2: Sort pts. arnd. p (in polar angle) in increasing order of angle.
Step 3: Walk arnd. to remove concave angle.
(keep pts. with left turns & drop those with right turns)
Step 4: Working arnd all pts. by sorting order.

* Divide & Conquer Approach

- Sort pts by x-coordinates
- For i/p set S,
  → divide into left half A & right half B using x-coords.
  → Compute CH(A)
  → " CH(B)
  → Combine
- Solve smaller problem using brute force.

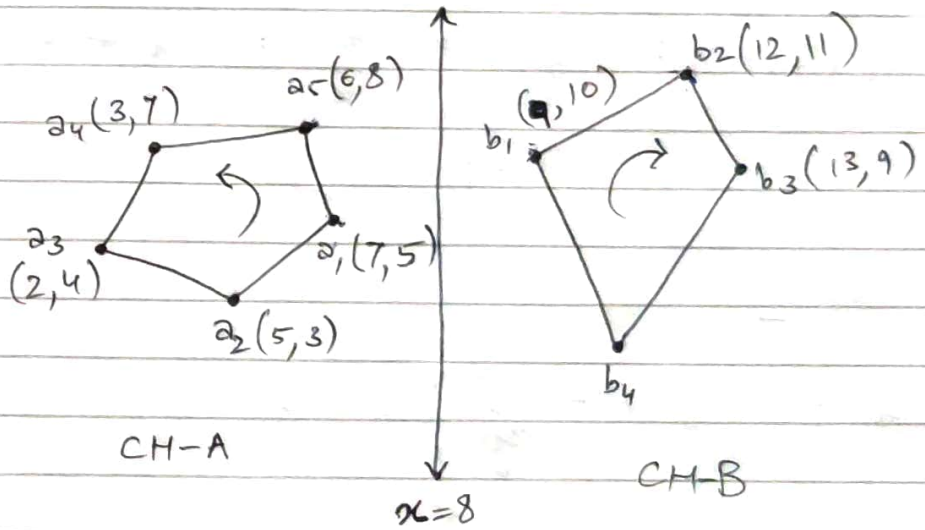
How to combine? Finding tangents
Brute force: generate pairwise (from A to B) segments & check



a_4, b_2 is called the upper tangent; Max $y(i,j)$
a_2, b_3 is called the lower tangent; Min $y(i,j)$
Complexity : $O(n^2)$

Picking max y for both isn't enough.
Need to maximise $y(i,j)$ for upper tangent & minimise in case of lower tangent.

- Two Finger Algo.:
  → Start with line seg. using rightmost of a & leftmost of b
  → Move cw for b → update if it improves $y(i,j)$.
  → Move ccw for a → update if it improves $y(i,j)$.
  → Repeat until $y(i,j)$ converges



$a_1 b_1$: $m = \frac{10-5}{9-7} = \frac{5}{2}$

$c = y_1 - mx_1 = 5 - 2.5(7)$

$= -12.5$

$y = 7.5$

a_1, b_2 : $(y - 5) = \frac{6}{5}(x - 7)$

$y - 5 = \frac{6}{5} \Longrightarrow y = 6.2$

$a_5(6,8)$ $b_1(9,10)$: $y - 8 = \frac{2}{3}(x - 6)$

$y = 8 + \frac{4}{3} = 9.33$

$y_{max}(i,j) = y(5,1) = 9.33 \Longrightarrow$ Set $a_5 b_1$

$a_5(6,8)$ $b_2(12,11)$: $y - 8 = 0.5(x - 6)$

$y = 9$

set $a_5 b_1$

a_4 b_1 : $y - 7 = \frac{5}{8}(x - 3)$

$= 2.5$

$y = 9.5$

Set $a_4 b_1$

$a_4(3,7)\ b_2(12,11)$: $y-7 = \frac{4}{9}(x-3)$

$$y = \frac{20}{9} + 7$$

$$= \frac{83}{9} = 9.2\ldots$$

set $a_4 b_1$

$a_3(2,4)\ b_1(9,10)$: $y-4 = \frac{6}{7}(x-2)$

$$y = \frac{36+28}{7}$$

$$= \frac{64}{7} = 9.1\ldots$$

Set $a_4 b_1$

$a_4 b_2$: $\Longrightarrow$ conversion

$\Longrightarrow$ tangent : $a_4 b_1$

Exp. 2B

Convex Hull

Set of pts. : $\{(1,1), (2,4), (3,2), (5,3), (4,1)\}$

Orientation(pt a, pt b, pt c) = $\begin{vmatrix} a.x & b.x & c.x \\ a.y & b.y & c.y \\ 1 & 1 & 1 \end{vmatrix}$

1) Brute

For line $(1,1) \rightarrow (2,4)$: or $((1,1), (2,4), (3,2)) = -1 \Rightarrow$ CCW

or $((1,1), (2,4), (5,3)) = -1 \Rightarrow$ CCW

or $((1,1), (2,4), (4,1)) = 1 \Rightarrow$ CW

∴ All pts. don't lie on the same side of the line, the line seg. can't be part of the hull.

2) Graham's Scan

Pt. with lowest y-coord : (1,1) ..... {∵ x-coord of pt. (1,1) is less than that of pt. (4,1)}

Now, we need to sort all the pts. by polar angle w.r.t. anchor pt.

⟹ Sorted pts.: [(1,1), (4,1), (3,2), (5,3), (2,4)]

Build Hull: Start with {(1,1)}

Add (4,1) ⟹ hull = {(1,1), (4,1)}

" (3,2) ⟹ check orientation((1,1), (4,1), (3,2)) = −1 ⟹ CCW

⟹ Part of the hull ⟹ hull: {(1,1), (4,1), (3,2)}

Add (5,3) ⟹ check or((4,1), (3,2), (5,3)) = −1 ⟹ add to hull

Add (2,4) ⟹ check or((3,2), (5,3), (2,4)) = −1 ⟹ CCW

But check or((4,1), (3,2), (2,4)) = 1 ⟹ CW

∴ pop (3,2) ⟹ check or((1,1), (4,1), (2,4)) = −1

so, add to hull ⟹ final hull: {(1,1), (4,1), (5,3), (2,4)}

3) Divide & Conquer

Divide:
Left = [(1,1), (2,4)]
Right = [(3,2), (4,1), (5,3)]

Conquer:
Compute left hull ⟹ ∵ only two pts, hull = {(1,1), (2,4)}
" right " ⟹ for 3 pts. we check their orientations.
or ((3,2), (4,1), (5,3)) = -1
or ((5,3), (3,2), (4,1)) = -1
or ((4,1), (5,3), (3,2)) = -1
All CCW ⟹ hull = {(3,2), (4,1), (5,3)}

Merge: Find rightmost pt. in left hull : (2,4)
Find leftmost " " right " : (3,2)
" upper tangent : (2,4) → (5,3)
" lower " : (1,1) → (4,1)
Merged hull = {(1,1), (4,1), (5,3), (2,4)}