# Sorting

↳ exchange argument

1. $A = \langle 3,5,7,1 \rangle$
   $B = \langle 2,8,4,5 \rangle$ → min value of $(A \cdot B)$ given reordering possible.

   ↳ $A \cdot B = 6 + 40 + 28 + 5 = 79$

# logic

$A = \langle 1,3,5,7 \rangle$
$B = \langle 8,5,4,2 \rangle$

$A \cdot B = 8 + 15 + 20 + 14 = 57$ → could work for only $\geq 0$ inputs.

* $1 \leq |A|, = |B| \leq 10^5$
$-10^9 \leq A_i, B_i \leq 10^9$

What for -ve numbers?

A → < -ve set > ; < +ve set >
B → < -ve set > ; < +ve set >

$\langle -3, -5 \rangle \quad \langle 5, 6, 7 \rangle$
$\langle -2, -1 \rangle \quad \langle 1, 2, 3 \rangle$ sorted

i) pair the most -ves with most +ve
↓
ii) It happen automatically for [a] reverse sorted and [b] sorted & then a·b

## Exchange argument :

$A \rightarrow \langle a_1, a_2, \ldots, a_n \rangle$ (suffle) } obs. 1
$B \rightarrow \langle b_1, b_2, \ldots, b_n \rangle$ (fixed) $\longrightarrow$ sorted.

suppose

$A \rightarrow \langle a_1 \; a_2 \ldots a_i \ldots a_j \ldots a_n \rangle$ ] be optimal
$B \rightarrow \langle b_1 \; b_2 \ldots b_i \ldots b_j \ldots b_n \rangle$ ] min (A·B)

$A \cdot B \rightarrow a_1b_1 + a_2b_2 \ldots + a_ib_i + \ldots a_jb_j + \ldots + a_nb_n$

**proof :**

After exchanging $a_i$ & $a_j$

$A \cdot B \rightarrow a_1b_1 + a_2b_2 + \ldots + a_jb_i + \ldots + b_ja_i + \ldots + a_nb_n$

Now,

$$a_ib_i + a_jb_j \times a_jb_i + b_ja_i$$

$$\Rightarrow a_j(b_i - b_j) + a_i(b_j - b_i) \geq 0$$

$$\Rightarrow (b_j - b_i)(a_i - a_j) \geq 0$$

$\because$ b was sorted, $b_j > b_i$

$$\therefore \underbrace{a_i - a_j > 0 \quad (a_i > a_j)}$$

↳ for every optimal sol$^n$

$\therefore$ A → should be sorted in reverse order.

this proof is actually valid for all $a_i, b_i \in \mathbb{R}$, not only -ve or +ve

$\therefore$

\# A → reverse sort

\# B → sort

take A·B

2. Codeforces

N questions

↳ $(P_1 \quad P_2 \quad P_3 \quad \ldots \quad P_N)$

Score $(S_1 \quad S_2 \quad S_3 \quad \ldots \quad S_N)$ → score of problems

Decay $(D_1 \quad D_2 \quad D_3 \quad \ldots \quad D_N)$ → score decay per unit time

Time to score $(T_1 \quad T_2 \quad T_3 \quad \ldots \quad T_N)$

order in which your should solve the questions so that you get max$^m$ total score let's take only 2 problem

$P_1$ $\quad$ $P_2$

$S_1 - D_1 T_1$ $\quad$ $S_2 - D_2(T_1 + T_2)$ } optimal

$P_2$ $\quad$ $P_1$

$S_2 - D_2 T_2$ $\quad$ $S_1 - D_1(T_1 + T_2)$

$$\cancel{S_1 + S_2} - D_1 T_1 - D_2(T_1 + T_2) > \cancel{S_1 + S_2} - D_2 T_2 - D_1(T_1 + T_2)$$

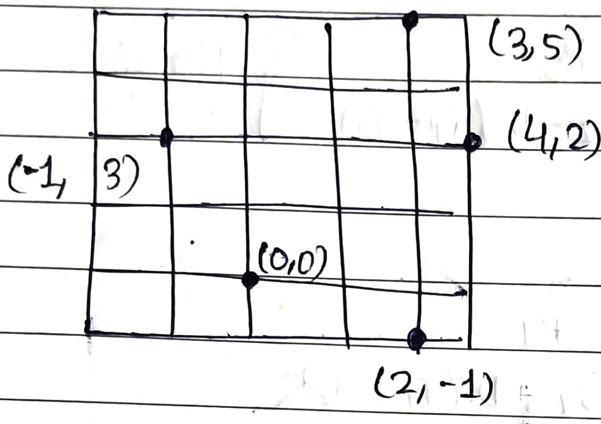
$$\Rightarrow \quad D_1 T_2 > D_2 T_1$$

$$\Rightarrow \quad \underbrace{\left(\frac{D_2}{T_2}\right) < \left(\frac{D_1}{T_1}\right)}_{\text{sorting method}}$$

sort by decreasing $\left(\frac{D_i}{T_i}\right)$

# Meet up problem :-

3.



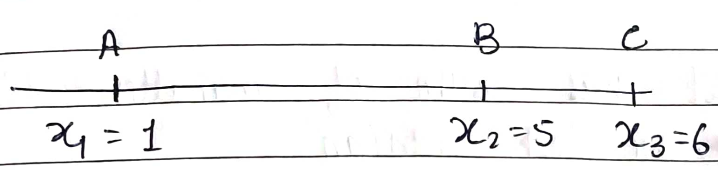

**Q.** In one move, you can move one person by one unit.
Find minimum moves for them to meet.

sol$^n$ Suppose they meet at $(x_0, y_0)$

$$\left\{ \begin{array}{l} |x_0+1| + |y_0-3| + |x_0-4| + |y_0-2| \\ + |x_0-2| + |x_0+1| + |x_0-3| + |y_0-5| \end{array} \right\} \text{minimize}$$

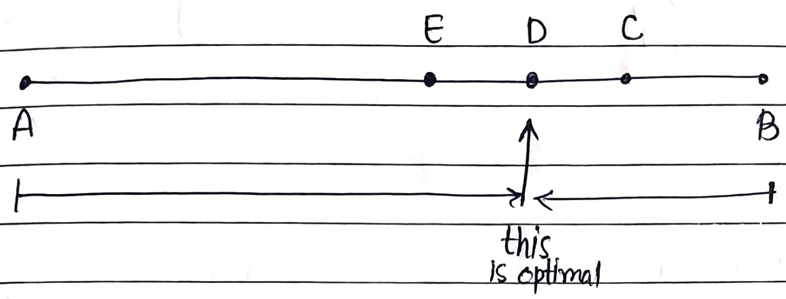
$$= \underbrace{\sum_{i=1}^{n} |x_0 - x_i| + \sum_{i=1}^{n} |y_0 - y_i|}_{\text{for this to be min}}$$

Date __ / __ / __




$$\underbrace{\sum_{i=1}^{n} |x_0 - x_i|}_{\min}$$

here they should meet ~~at~~ at B, not $\bar{x}$




$r_A + r_B = \text{const.} = AB$

$r_E + r_C = \text{const.} = EC$

$r_D = 0$

$\therefore$ always choose the middle point not mean.

$\therefore \sum_{i=1}^{n} |x_0 - x_i| \Big\}$ optimal for $x_0 =$ median of $\underline{x_i\text{s}}$

$\therefore$ for $\min \left( \sum_{i=1}^{n} |x_0 - x_i| + \sum_{i=1}^{n} |y_0 - y_i| \right)$

$\hookrightarrow \underline{(x_0, y_0) \equiv \text{median}(x_i\text{s}, y_i\text{s})}$

Here, cost was sum of mahatten distance,
so, we took median

If, cost was eucledian distance, then take mean.

$$Cost = \sum_{i=1}^{n} \sqrt{(x-x_i)^2 + (y-y_i)^2} \Big\} \text{ optimal } (x,y) = (\bar{x}, \bar{y})$$

4. N person team :

↳ ($S_1$ $S_2$ $S_3$ ...... $S_N$) (students)

($P_1$ $P_2$ $P_3$ ...... $P_N$) (Power)

($E_1$ $E_2$ $E_3$ ...... $E_N$) (efficiency)

choose k interns & make a team

↳ maximize $\left[\left(\sum P_i\right) \cdot \min(E_i)\right]$ $(1 \le i \le k)$

Constraints:

$N, K \le 10^5$ & $P_i, E_i \le 10^6$

Idea-#:

Sort in terms of e.

eg: p: 9 5 3 8 1 (k=3)

e: 1 3 5 7 8

1st option: (1+8+3)(5) = 60

2nd option: (5+3+8)(3) = 48

3rd option: (9+5+8)(1) =

take max

fix the min efficiency & take top k sums.

add (x)
sumOftopK () } priority queue / multiset

# Standard Problems :-

- Sorting
- Reversal of approach / Thinking backward
  $x \rightsquigarrow y$
- splitting Dimesion
- multiset / Priority Queue
- Range based Problems

# Activity Selection Problem :-

Given n job activities with their start and finish time. Find the maximum number of activities that can be performed without performing two activities at the same time.

Greedy idea : Select the job which end earliest. So, we will sort all the jobs with their end -time and greedily pick the jobs if it is not clashing with previous picked job.

# More Greedy Ideas :-

- → Counting.
- → Optimization.

→ (mostly intuitive)

**Reversal** Q. $X \to Y$ बनाना है। (Classical)

{ You can either go to $2X$ from $X$

→ op 1: $X \to 2X$

You can go to $X-1$ from $x$

→ op2 : $X \to X-1$

Min$^m$ ops to reach $y$. from $x$.

sol$^n$: उल्टा चलते हैं।

$Y \xrightarrow{op\,1} Y/2$ , if $y$ is even

$Y \xrightarrow{op\,2} (Y+1)$

Now, we can make some arguments

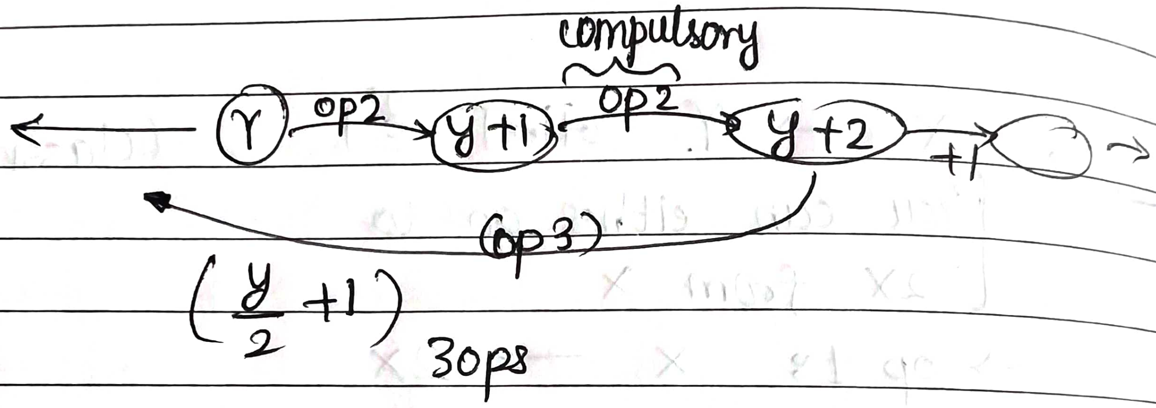
if $Y <= X \Rightarrow$ moves $+= (X - Y)$

if $Y > X \Rightarrow$

if (Y > X)

Case I: y is odd ⇒ $\underline{y+1}$

Case II: y is even ⇒




# But y can reach $\left(\frac{y}{2}+1\right)$ in 2 ops by

↳ (op 1) then (op 2)

↳ so, its always better to go back first & the then come to front.

# Substraction game

Q $X$, $A\{1, 2, 3, \ldots, y\}$

$P_1$, $P_2$ can choose any no. from set A & substract from x till X becomes < 0. player who makes $X < 0$ looses.
If $P_1$ plays first & both plays optimally, find who will win.

soln: if $X \% (y+1) == 0$, then you will loose
else You can always win.