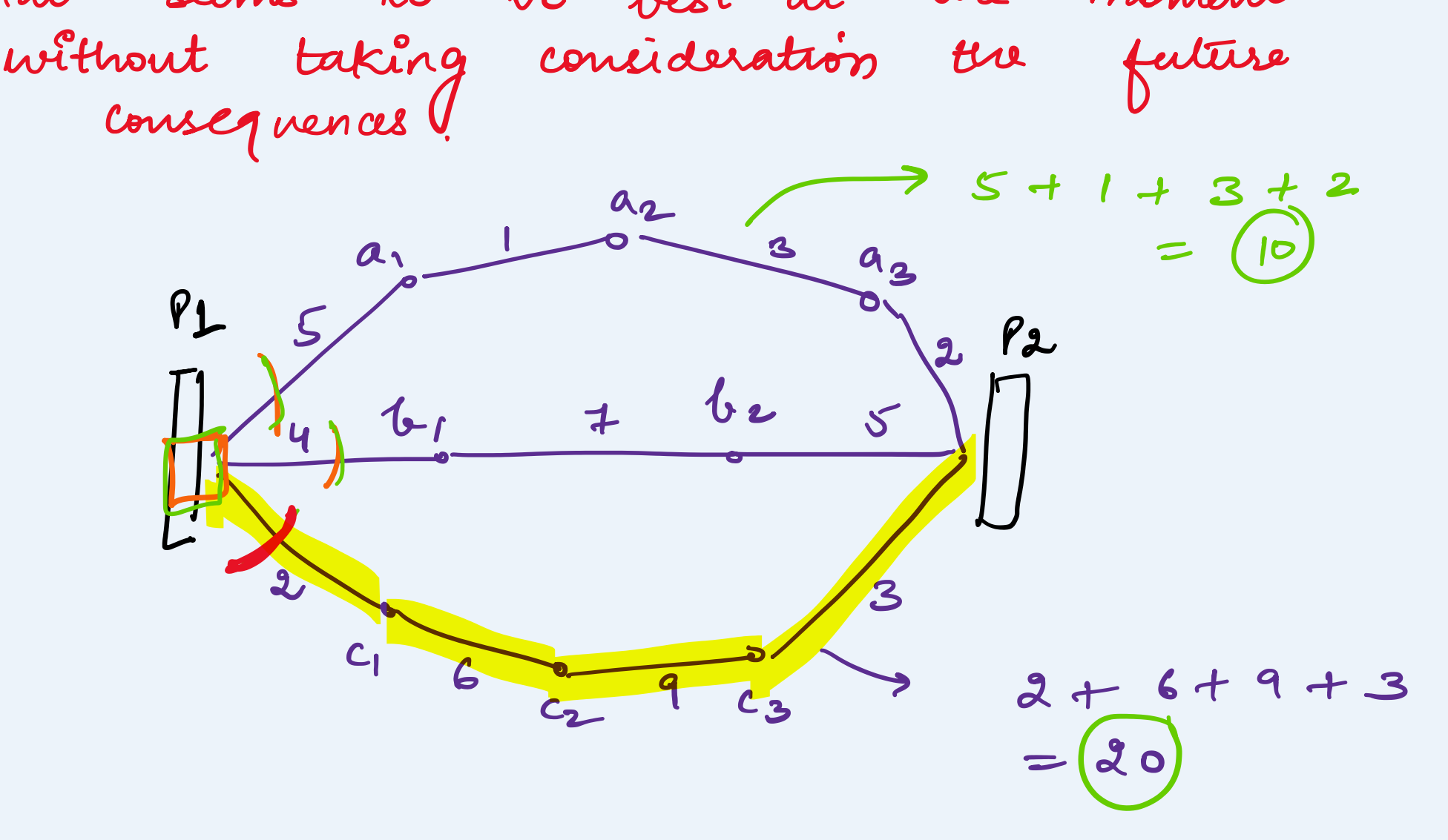
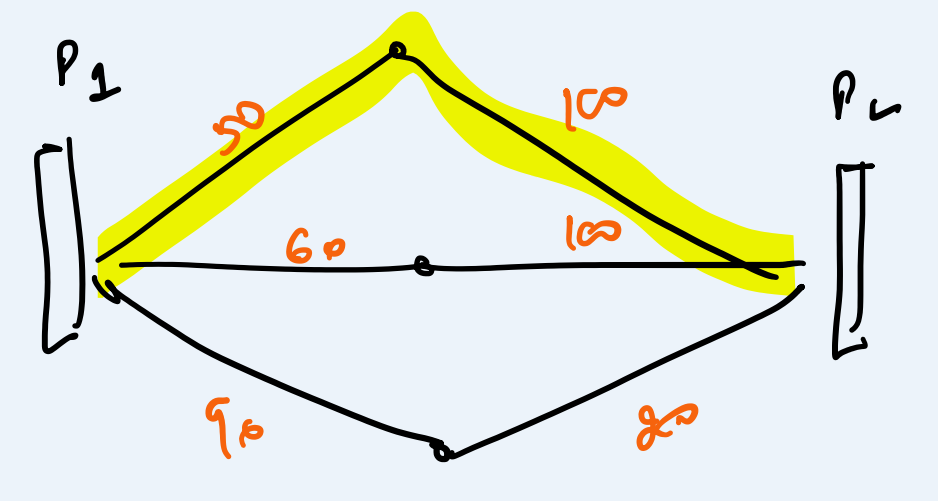
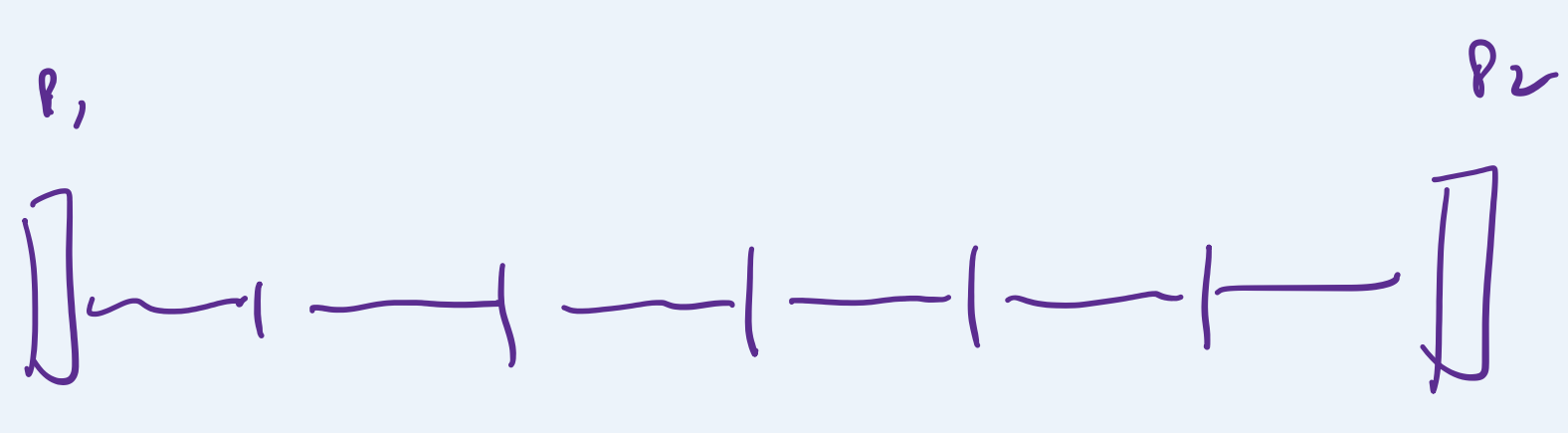
Monday, 25 October 2021 8:35 PM

Greedy Algorithms

Greedy algorithms always make the choice that seems to be best at the moment without taking consideration the future consequences.







1. Greedy algorithms are always based on intuition & observation.
2. Greedy algorithms can only work when local answer contributes to global answer.

Two properties:

① Greedy choice property

If by choosing the best choice at each step without reconsidering the previous step once chosen.

② optimal substructure

If the optimal overall solution to the problem corresponds to the optimal solution to its subproblems.

Q Task scheduler (Leetcode)

Ans: Total no. of CPU cycles = Busy + Idle

No. of busy slots = no. of tasks execute
= len (arr)

Max no. of idle slots is defined by the freq of most freq task.

A B A A B C A A , n = 2

A → ⑤

A _ _ A _ _ A _ _ A _ _ A
(2)

f_max = ⑤

Max possible idle time = (f_max − 1) × cooling period.
= (5 − 1) * 2
= 8

To minimise idle slots, execute other tasks in idle time.

A B A A B C A A

① Freq & sort
A → 5
→ B → 2
C → 1

A B C A B _ A _ _ A _ _ A   ⑧ − 3

Total time = 8 + 5 = ⑬ ✓

Q Given an array of +ve & −ve integer, find the min possible subset product.

Eg: {−1, −1, −2, 4, 3}   ans = −24
Eg: {−1, 0}   ans = −1
Eg: {−1, 0, 2}   ans = −2

what if all +ve no ?? [no negative no]
[2 1 3 5]
[2 1 3 4 0 5]
ans → min ele of the array

what if 1 −ve no present ??
[2 1 3 0 4 5 −1]
ans = −1 × (all other no except 0.)

what if 2 −ve no present ??
2 1 3 4 5 × (−2) −1
ans = −2 (−ve no with smaller magnitude) × (all positive except 0.)

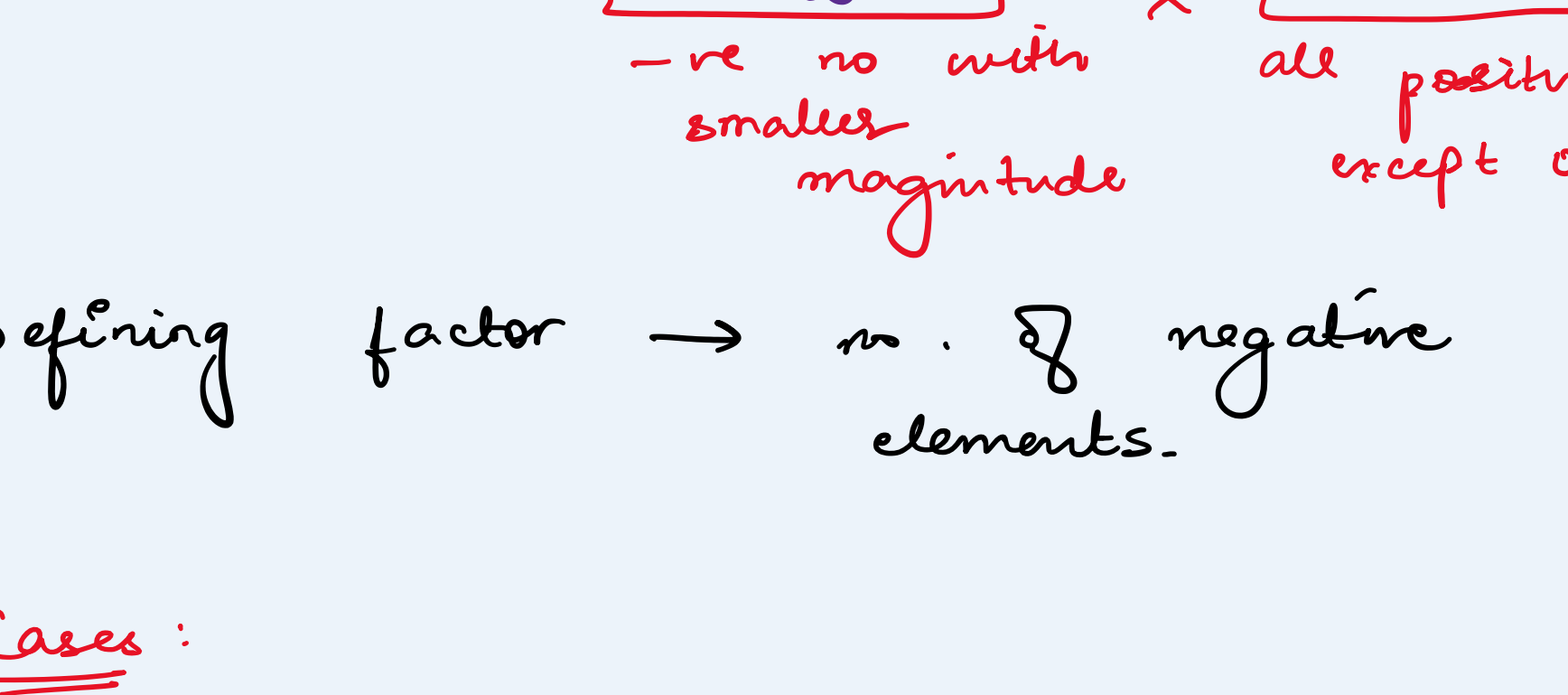
Defining factor → no. of negative elements.

Cases:

① no. of negative no → min (arr)
② even no. of negative no →
→ Π [a[i]] (a[i] ≠ 0) × negative no of min value
③ odd no. of negative no →
Π [a[i]]
a[i] ≠ 0

Q Mice & holes

mices → [4, −4, 2]
holes → [4, 0, 5]



ans depends on the max distance that any mice has to travel.

① Sort the mice positions
② Sort hole positions.

[mice[i] − hole[i]]

mice → [−4, 2, 4]
holes → [0, 4, 5]
4 2 1

i = 0   = |−4 − 0| = 4
i = 1   = |4 − 2| = 2
i = 2   = |4 − 5| = 1

max (4, 5, 1) = ④ (ans)