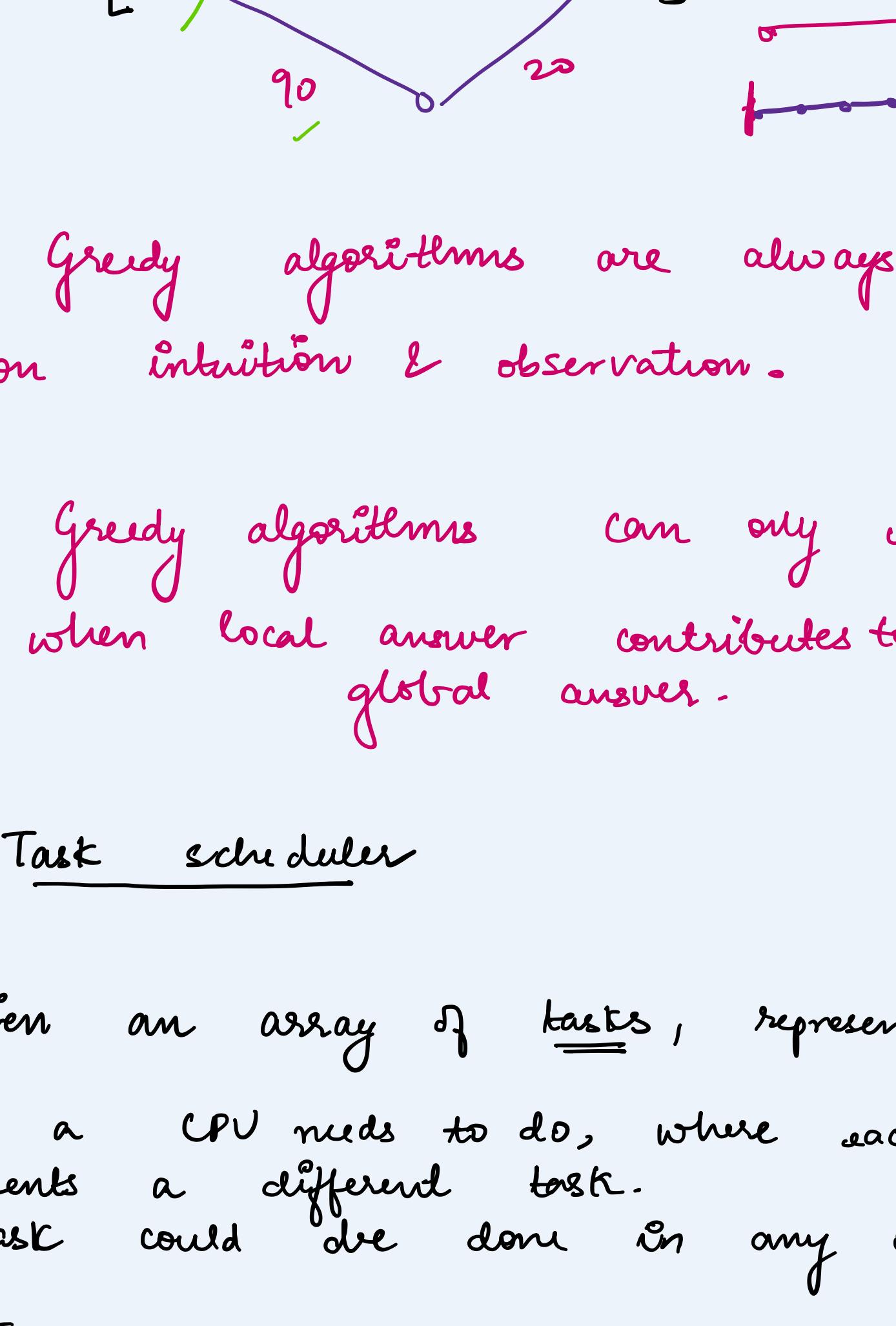
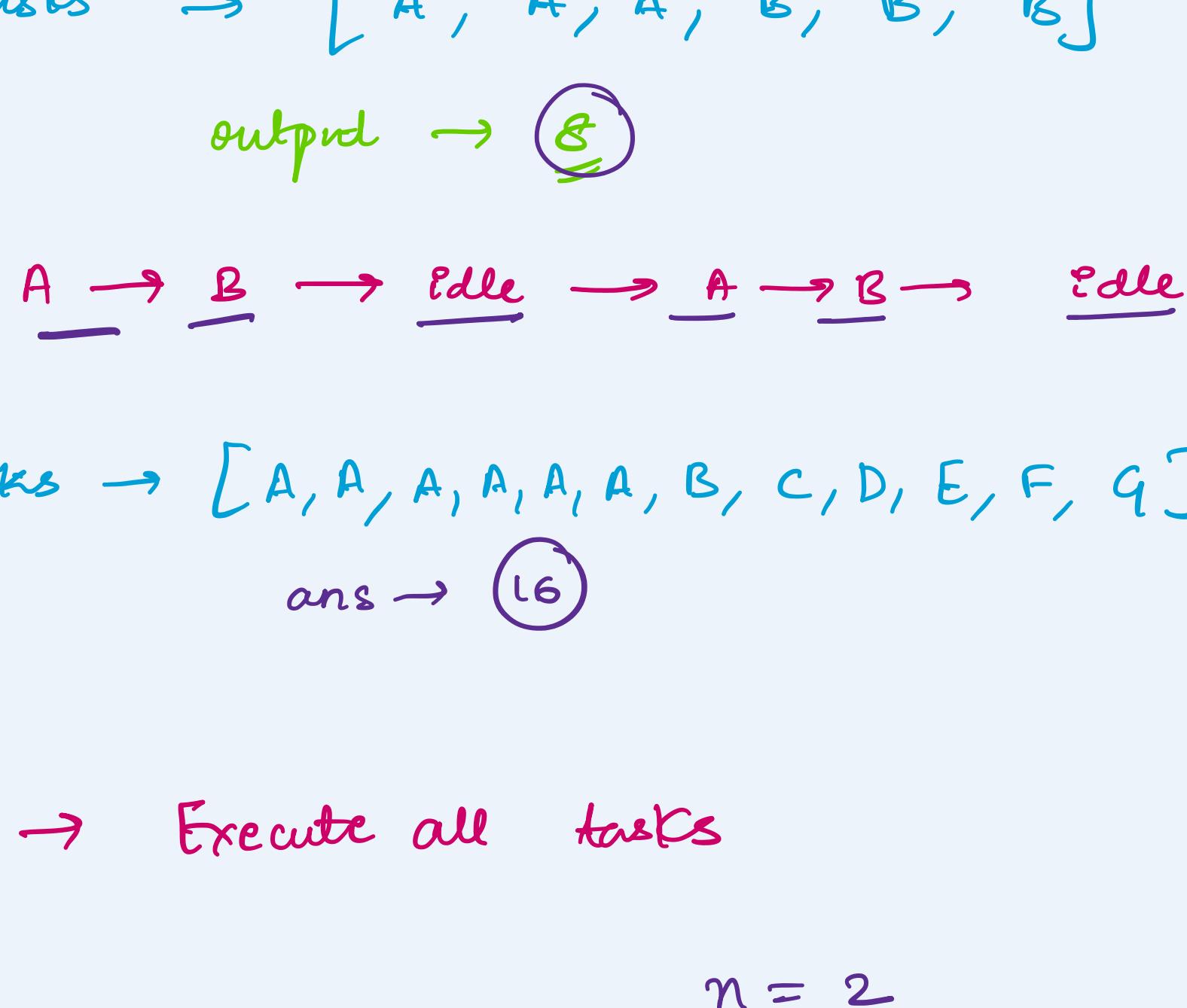


Greedy Algorithms

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



$$4 + 1 + 3 + 2 = 10$$



1. Greedy algorithms are always based on intuition & observation.

2. Greedy algorithms can only work on when local answer contributes to the global answer.

Task scheduler

Given an array of tasks, representing the

tasks a CPU needs to do, where each letter represents a different task.
→ Task could be done in any order.

- In 1 unit of time, CPU either → complete one task → idle.
- n → cooldown period b/w 2 tasks
↳ i.e., there must be atleast n units of time b/w any two same tasks.
- least the number of units of times that CPU will take to finish all given tasks.

(1) tasks → [A, A, A, B, B, B] n=2
output → ⑧

output A → B → idle → A → B → idle → A → B

(2) tasks → [A, A, A, A, A, B, C, D, E, F, G] n=2
ans → ⑯

→ Execute all tasks

$$n = 2$$

A — — A

Final ans → busy + idle slots
↓
no of tasks to execute
↓
len(arr)

May be possible idle slots is going to be dependent on frequency of most frequent task.

A B A A B C A A n=2

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

$$\text{max idle-time} = (f - \text{max} - 1) * n$$

$$= (5 - 1) * 2$$

$$= 8$$

A → 5 B → 2 C → 1

8 + 5 = 13

n=2

A A B B C C

6 + 1

6

0

A — — A

⑩

Given array, consisting of +ve & -ve integers. 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??

→ [2 1 3 4 5]

ans → min no.

→ [2 1 3 0 4 5]

ans → -1 × all other

no, except 0

→ what if 2 -ve no. ??

→ [2 1 3 4 5 -2] -1

→ choose all the +ve no and -ve no. with greater magnitude.

→ [-2 1 3 4 5] -1

→ [-2 1 3 4 5] -1

→ [-2 1 3 4 5] -1

→ [-2 1 3 4 5] -1

→ [-2 1 3 4 5] -1

→ [-2 1 3 4 5] -1

→ [-2 1 3 4 5] -1

→ [-2 1 3 4 5] -1

→ [-2 1 3 4 5] -1

→ [-2 1 3 4 5] -1

→ [-2 1 3 4 5] -1

→ [-2 1 3 4 5] -1

→ [-2 1 3 4 5] -1

→ [-2 1 3 4 5] -1

→ [-2 1 3 4 5] -1

→ [-2 1 3 4 5] -1

→ [-2 1 3 4 5] -1

→ [-2 1 3 4 5] -1

→ [-2 1 3 4 5] -1

→ [-2 1 3 4 5] -1

→ [-2 1 3 4 5] -1

→ [-2 1 3 4 5] -1

→ [-2 1 3 4 5] -1

→ [-2 1 3 4 5] -1

→ [-2 1 3 4 5] -1

→ [-2 1 3 4 5] -1

→ [-2 1 3 4 5] -1

→ [-2 1 3 4 5] -1

→ [-2 1 3 4 5] -1

→ [-2 1 3 4 5] -1

→ [-2 1 3 4 5] -1

→ [-2 1 3 4 5] -1

→ [-2 1 3 4 5] -1

→ [-2 1 3 4 5] -1

→ [-2 1 3 4 5] -1

→ [-2 1 3 4 5] -1

→ [-2 1 3 4 5] -1

→ [-2 1 3 4 5] -1

→ [-2 1 3 4 5] -1

→ [-2 1 3 4 5] -1

→ [-2 1 3 4 5] -1

→ [-2 1 3 4 5] -1

→ [-2 1 3 4 5] -1

→ [-2 1 3 4 5] -1

→ [-2 1 3 4 5] -1

→ [-2 1 3 4 5] -1

→ [-2 1 3 4 5] -1

→ [-2 1 3 4 5] -1

→ [-2 1 3 4 5] -1

→ [-2 1 3 4 5] -1

→ [-2 1 3 4 5] -1

→ [-2 1 3 4 5] -1

→ [-2 1 3 4 5] -1

→ [-2 1 3 4 5] -1

→ [-2 1 3 4 5] -1

→ [-2 1 3 4 5] -1

→ [-2 1 3 4 5] -1

→ [-2 1 3 4 5] -1

→ [-2 1 3 4 5] -1

→ [-2 1 3 4 5] -1

→ [-2 1 3 4 5] -1

→ [-2 1 3 4 5] -1

→ [-2 1 3 4 5] -1

→ [-2 1 3 4 5] -1

→ [-2 1 3 4 5] -1

→ [-2 1 3 4 5] -1

→ [-2 1 3 4 5] -1

→ [-2 1 3 4 5] -1

→ [-2 1 3 4 5] -1

→ [-2 1 3 4 5] -1

→ [-2 1 3 4 5] -1

→ [-2 1 3 4 5] -1

→ [-2 1 3 4 5] -1

→ [-2 1 3 4 5] -1

→ [-2 1 3 4 5] -1

→ [-2 1 3 4 5] -1

→ [-2 1 3 4 5] -1

→ [-2 1 3 4 5] -1

→ [-2 1 3 4 5] -1

→ [-2 1 3 4 5] -1

→ [-2 1 3 4 5] -1

→ [-2 1 3 4 5] -1

→ [-2 1 3 4 5] -1

→ [-2 1 3 4 5] -1

→ [-2 1 3 4 5] -1

→ [-2 1 3 4 5] -1

→ [-2 1 3 4 5] -1

→ [-2 1 3 4 5] -1

→ [-2 1 3 4 5] -1

→ [-2 1 3 4 5] -1

→ [-2 1 3 4 5] -1

→ [-2 1 3 4 5] -1

→ [-2 1 3 4 5] -1

→ [-2 1 3 4 5] -1

→ [-2 1 3 4 5] -1

→ [-2 1 3 4 5] -1

→ [-2 1 3 4 5] -1

→ [-2 1 3 4 5] -1

→ [-2 1 3 4 5] -1

→ [-2 1 3 4 5] -1

→ [-2 1 3 4 5] -1

→ [-2 1 3 4 5] -1

→ [-2 1 3 4 5] -1

→ [-2 1 3 4 5] -1

</div