

## Today's Content

Painters Pavilion  
Aggressive cows.

## Today's Quote

A dream without clarity is  
a nightmare.

## Painters Partition Problem.

Ques

We have to paint  $n$  boards of length  $A_1, A_2, \dots, A_n$ . There are  $k$  painters available and each takes 1 unit of time to paint 1 unit of board.

find min time to get the job done.

→ 1 painter will paint only continuous sections of the board.

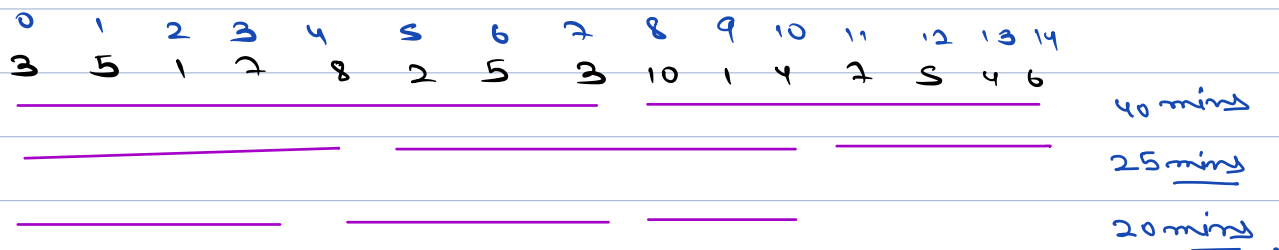
→ Two painters can't share a board.

e.g. 1)  $A = \{ \overset{0}{\underline{10}}, \overset{1}{\underline{10}}, \overset{2}{\underline{10}}, \overset{3}{\underline{10}} \}$  ,  $A_n = \underline{20}$  ,  
 $k = 2$

e.g. 2)  $A = \{ \overset{0}{\underline{10}}, \overset{1}{\underline{20}}, \overset{2}{\underline{30}}, \overset{3}{\underline{40}} \}$  ,  $A_n = \underline{60}$  ,  
 $k = 2$

Ex  
N=15

K=3



idea 1:- divide the work into  $\frac{\text{totaltime}}{K}$

arr[6] = 1 1 1 2 1 100 K=2

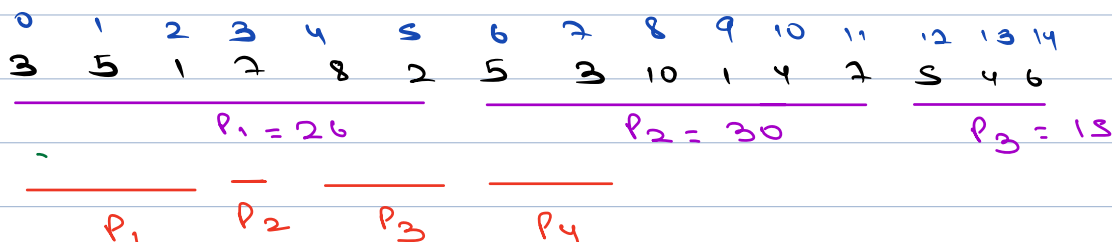
$$\frac{\text{sum}}{K} \Rightarrow \frac{106}{2} = 53$$

Ans = 100

Not possible to divide like this

Ex  
N=15

K=4



Can we do all task in 30 mins

30 31 32

Can we finish all tasks in 10 mins?

8 9 10  
x x x

8 9 10 . . x . . . 30 31 32  
x x x ✓ ✓ ✓ ✓

Target :- min time to paint all the boards

Search Space :-  
lo  $\rightarrow$  max(Array)  
hi  $\rightarrow$  sum(Array)

Ex  
N=15

k=4

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14  
3 5 1 7 8 2 5 3 10 1 4 7 5 4 6

$w_1 = 16$

$w_2 = 18$

$w_3 = 15$

$w_4 = 16$

$w_1 = 16$

$w_2 = 18$

$w_3 = 22$

$w_4 = 15$

$w_1 = 16$

$w_2 = 18$

$w_3 = 15$

$w_4 = 16$

$w_1 = 16$

$w_2 = 15$

$w_3 = 14$

$w_4 = 16$

$w_1 = 24$

$w_2 = 21$

$w_3 = 20$

$w_4 = 6$

$w_1 = 34$

$w_2 = 37$

lo

hi

mid

10

71

40

$Ans = 40$  goto left

10

39

24

$Ans = 24$  goto left

10

23

16

goto right

17

23

20

goto right

21

23

22

$Ans = 22$  goto left

21

21

21

goto right

22

21

Break

Pseudo Code :-

```
int workers (int time[], int n, int k) {  
    int lo = maxEle(arr);  
    int hi = sum(arr);  
    while (lo <= hi) {  
        m = lo + (hi - lo) / 2;  
        if (check(m, time, k)) {  
            ans = m;  
            hi = mid - 1;  
        }  
        else {  
            lo = mid + 1;  
        }  
    }  
    return ans;  
}
```

T.C  $\rightarrow O(\log(\text{sum} - \text{max}) * n)$

S.C  $\rightarrow O(1)$

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
3	5	1	7	8	2	5	3	10	1	4	7	5	4	6

bool check (int m, int time[], int n, int k){

Δ = 0; c = 1,

for (i = 0; i < n; i++) {

Δ = Δ + time[i];

if (Δ > m) {

c++;

Δ = time[i];

if (c > k) {

return false;

}

}

}

return True;

}

## Aggressive Cows

Ques) Given  $N$  cows &  $M$  stalls, all  $M$  stalls are on  $x$  axis at diff. locations, place all  $N$  cows such a way that min dist b/w any 2 cows is maximized.

Note 1:- In a stall only one cow can be present.

Note 2:- All cows have to be placed.

	0	1	2	3	4	
<u>Ex 1:-</u>	1	2	4	8	9	Min dist
	$c_1$	$c_2$	$c_3$			1
Stalls = 5	$c_1$		$c_2$		$c_3$	3
Cows = 3	$c_1$			$c_2$	$c_3$	1



Ex 2

Stalls = 9

cows = 4

0	1	2	3	4	5	6	7	8
2	6	11	14	19	25	30	39	43
$c_1$	$c_2$	$c_3$	$c_4$					3
$c_1$		$c_2$		$c_3$		$c_4$		8
$c_1$			$c_2$			$c_3$	$c_4$	12

Stalls = 9

cows = 4

0	1	2	3	4	5	6	7	8
2	6	11	14	19	25	30	39	43
$c_1$					$c_2$			
$c_1$		$c_2$		$c_3$	$c_4$			

check if we can

place cows at  
min 20 distance

away.

→ 20 21 22 23  
- x x x

3 4 5 6 7 8 9 10 11 12  
✓ ✓ ✓ - x x x x x x

Can we place cows atleast 5 dist. away?

3 4 5  
✓ ✓ ✓

Target  $\rightarrow$  min dist b/w any two cows  
should be max.

Search space  $\rightarrow$  lo  $\rightarrow$  min dist b/w two adj. element,  
hi = max - min

cows  $\geq 4$ .

0	1	2	3	4	5	6	7	8
2	6	11	14	19	25	30	39	43

$C_1$  —————  $C_2$

$C_1$  —————  $C_2$  —————  $C_3$  —————  $C_4$

$C_1$  —————  $C_2$  —————  $C_3$

$C_1$  —————  $C_2$  —————  $C_3$

$C_1$  —————  $C_2$  —————  $C_3$

lo	hi	mid	
3	41	22	goto left

3	21	12	Ans = 12 goto right.
---	----	----	----------------------

13	21	17	goto left
----	----	----	-----------

13	16	14	goto left
----	----	----	-----------

13	13	13	goto left.
----	----	----	------------

13	12	break	
----	----	-------	--

Ans → 12.

Pseudo Code :-

```
int mo0 (int dist[], int n, int c) {  
    lo = min (dist[i+1] - dist[i]) * i;  
    hi = arr[n-1] - arr[0];  
    while (lo <= hi) {  
        m = lo +  $\frac{(hi-lo)}{2}$   
        if (check (m, dist, c)) {  
            ans = m;  
            lo = mid+1;  
        }  
        else {  
            hi = mid-1;  
        }  
    }  
    return ans;  
}
```

0	1	2	3	4	5	6	7	8
2	6	11	14	19	25	30	39	43
=				↓				
				last-placed				

```
bool check (int m, int dist[], int n,  
            int cows)
```

```
    last_placed = dist[0],  
    count = 1;  
    for (i = 1; i < n; i++) {  
        if (dist[i] - last_placed >= m) {  
            count++;  
            last_placed = dist[i];  
            if (count == cows) {  
                return true;  
            }  
        }  
    }  
    return false;
```

3

Task to do

1- 2 constraint

(k painters)

(c cows)

1 minimize or maximize.

Search Space :- whatever we are asked to  
minimize or maximize.

Binary search on Answer.

T T T T T T T F F F F F (Agg. cows)

F F F F F F F T T T T T T T (Painters Partition)

monotonic nature.

→ Problem is feasible till a point  
and not after that  
or  
vice versa.

$$x^2 \leq n$$

↙  
longest n,

longest

so

floor(sqrt)n

1	2	3	4	5	6	7	8	9	10	10
T	T	T	T	T	T	T	F	F	F	F

category,

5<sup>th</sup>, 2,3

1	2	3	4	5	6	7	8	9	10	X	X	X
										T	F	F

x at which the no. of  
we ≤ s,

— Block Allocation Problem —

$$\begin{matrix} \rightarrow \\ \rightarrow \\ \rightarrow \end{matrix} \begin{bmatrix} 7 & 9 \\ 2 & 1 \\ 4 & 9 \end{bmatrix} \quad \underline{3 \times 2}.$$

$$\begin{matrix} & 1 & 2 & 3 \\ 1 & 4 & 5 & 6 \\ 2 & 7 & 8 & 9 \end{matrix} \Rightarrow \begin{matrix} \overset{1}{\underbrace{1, 5, 9}} \\ \underbrace{2, 7, 8} \end{matrix}$$

(1, 4, 9)  
3 5



