

08/10/2026  
Sunday

## Allocate Books

Given an array

$a[i]$  represents no of pages in the  
ith book

There are  $m$  no of stds the task is to  
allocate the all books to the students

Allocate books in such a way

1. Each student gets atleast one book

2. Each book should be allocated to  
only one student

3. Book allocation should be in contiguous  
ranges if not possible return -1

Ex: we need to find min(max)

Given

$a = [25 \ 46 \ 28 \ 49 \ 24]$

(m) stds = 4

I know one thing if no of books are  
less than the students then it is not  
possible.

Suppose

std 1 - 25

std 2 - 46

" 3 - 28

4 - 49

5 - 24

but we need to assign  
all to  $m$  stds  
only

1 - 25

2 - 25 we cannot add bcz of cond<sup>n</sup> 2

1 - 25

2 - 46

3rd cond<sup>n</sup>

should be in contiguous

1 - 25 + 46

2 - 28

3 - 49

4 - 24

1 way at  $25, 46 | 28 | 49 | 24$

2 ways at  $25 | 46 + 28 | 49 | 24$

3 ways at  $25 | 46 | 49 + 28 | 24$

4 ways at  $25 | 46 | 28 | 49 + 24$

we need to find min(max)

among all this min is 71

take another example

[25 46 28 49 24]

if only 1 stat then I can

allocate all books so

max is sum(a)

high is sum(a)

and low is max(a)

I have to max pages  
book of

$$\begin{array}{ccccccc} \text{low} & & \text{mid} & & \text{high} \\ 49 & & (172+49) & \cancel{\frac{221}{2}} & 172 \\ & & 110 & & \end{array}$$

for 110, I will find how many states  
I can get

$$\text{stttt } 1 - 25 + 46 + 28 < 110$$

Stt 2 - 49+24 410

only 2 stats but we need 4 stats  
in

So we should go left side

high = mid - 1

law  
49

mid

night

99

$$\frac{49+99}{7}$$

74

Now

I will check for ~~74~~

$$\text{Start} \ 1 - 25 + 46$$

~~CHI~~ 2-28-~~19~~

~~Open~~ ~~3-49+24~~

but we need eight so

we go left side  $\leftarrow$  ~~high~~ = mid - 1

low

mid

~~high~~

49

$$\frac{49+73}{2}$$

et of Sat at 61 Grade & Rd 08

for 6

soft 1 → 25

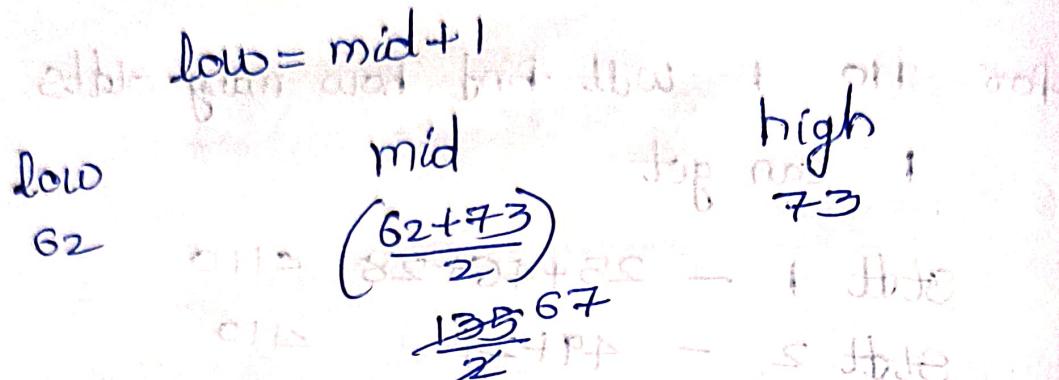
Stoff 2 → 4G

gktt 3 → 28

Statt 4 → 49

Stt 5 → 24

we got 5 students  
but we need 4 students only  
so we go to right side only



edit for 67

Stt 1  $\rightarrow$  25

Stt 2  $\rightarrow$  46

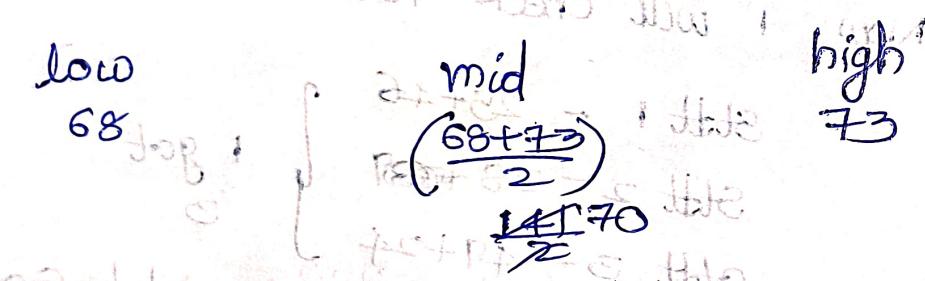
Stt 3  $\rightarrow$  28

Stt 4  $\rightarrow$  49

Stt 5  $\rightarrow$  24

still 5 students we need 4 students  
only so we go right side

low = mid + 1



Stt 1  $\rightarrow$  25

Stt 2  $\rightarrow$  46

Stt 3  $\rightarrow$  28

Stt 4  $\rightarrow$  49

Stt 5  $\rightarrow$  24

so all 5 stt's so we go to  
right side

low = mid + 1

low 71      mid  $\left(\frac{71+73}{2}\right)$  72      high 73

for 72

$$\text{stt 1} \rightarrow 25 + 46$$

$$\text{stt 2} \rightarrow 28$$

$$\text{stt 3} \rightarrow 49$$

$$\text{stt 4} \rightarrow 24$$

we got 4 students but we need min  
 right go we go left side

$$\text{high} = \text{mid} - 1$$

low 71      mid 72      high 73

for 71

$$\text{stt 1} \rightarrow 25 + 46$$

$$\text{stt 2} \rightarrow 28$$

$$\text{stt 3} \rightarrow 49$$

$$\text{stt 4} \rightarrow 24$$

we got 4 still we go right to  
 check any min of these

but low crosses high

high 70      low 71

so answer is low 70

high > low

function countstats (int a[], int m)

{

stats = 1;

noofpages = 0;

for (int i=0; i<a.length; i++)

{

if (noofpages + a[i] <= m)

{

noofpages += a[i];

else {

stats += 1;

noofpages = a[i];

}

y

return stats;

}

public int solve (int a[], int n, int m)

{

int low = max(a, n);

int high = sum(a, n);

while (low <= high) {

int mid = low + (high - low) / 2;

int countstats = countstats(a, mid);

if (countstats <= m)

{

high = mid - 1;

}

else {

low = mid + 1;

}

y

return low;

g

### painter's partition

$$a = [10 \ 20 \ 30 \ 40] \quad k=2$$

Given  $k$  painters  
these painters should paint in contiguous manner

10 units of area Each painter should get atleast one unit to paint

for  $k=2$

$$\begin{array}{ccccc} [10] & [20 \ 30 \ 40] & \text{max is } 90 \\ P_1 & P_2 & \frac{10}{90} \end{array}$$

$$\begin{array}{ccccc} [10 \ 20] & [30 \ 40] & \text{max is } 70 \\ P_1 & P_2 & \frac{30}{70} \end{array}$$

$$\begin{array}{ccccc} [10 \ 20 \ 30] & [40] & \text{max is } 60 \\ P_1 & P_2 & \frac{40}{60} \end{array}$$

among all this minimum is 60

Split array - Largest sum

↳ split the array into  $k$  subarrays such that the max subarray sum is minimum

it is also same as  
painter's partition

