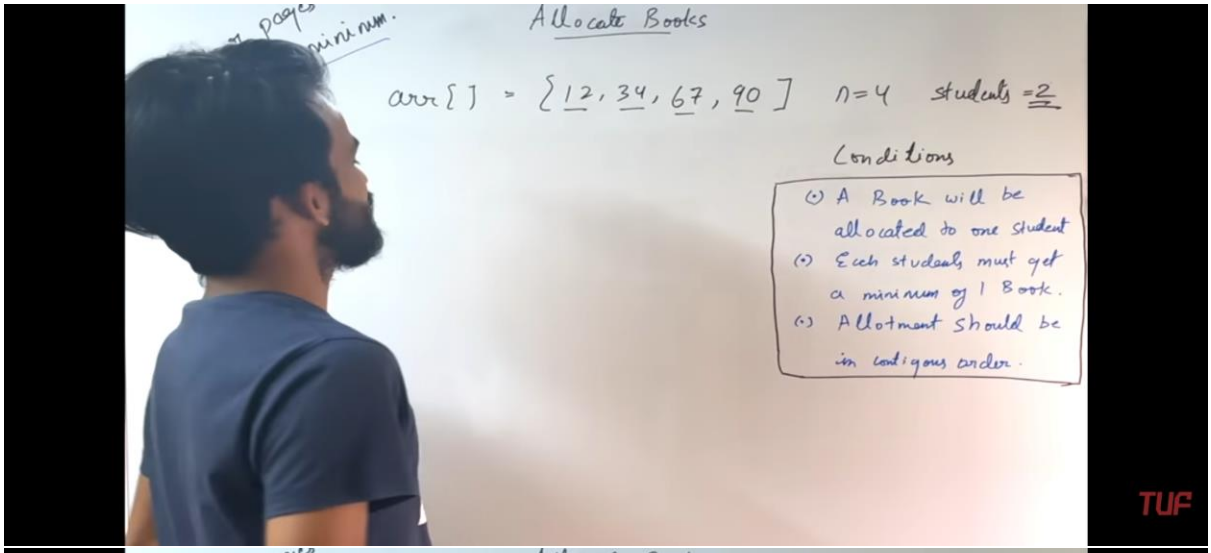
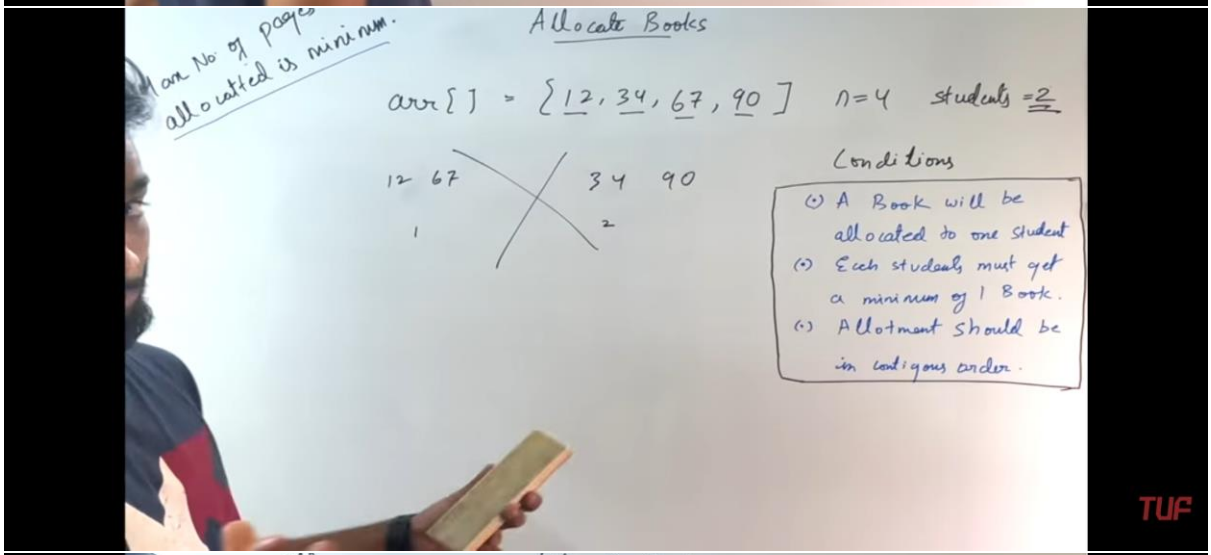
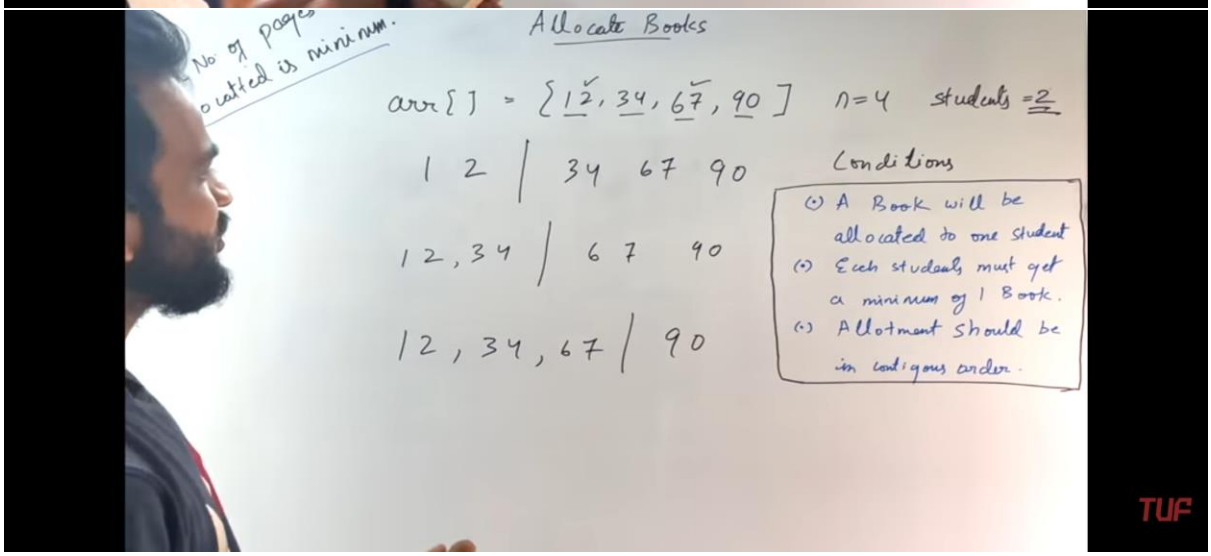
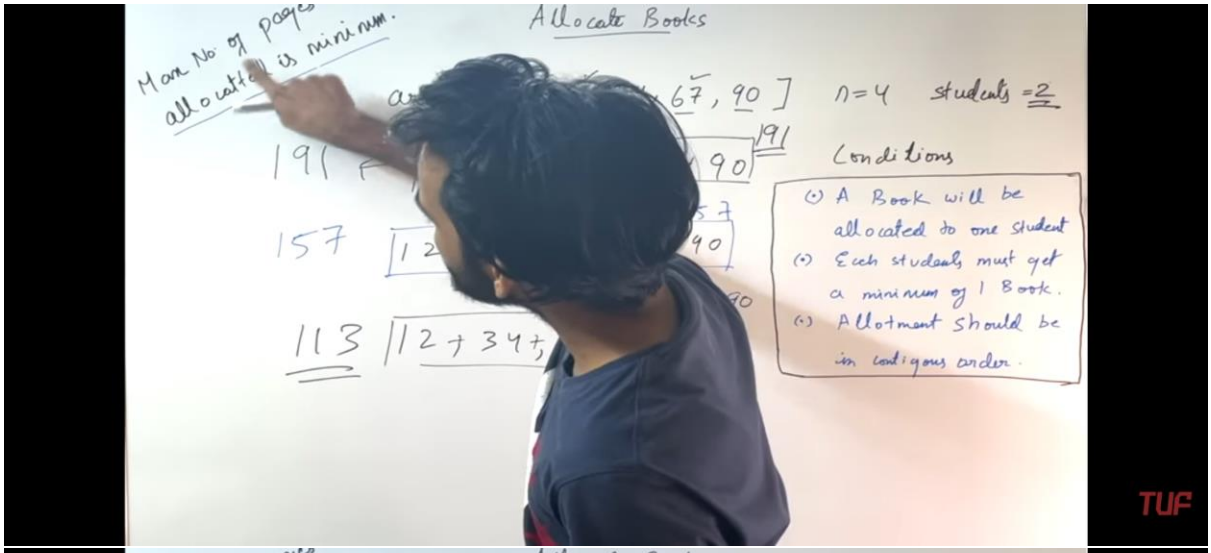
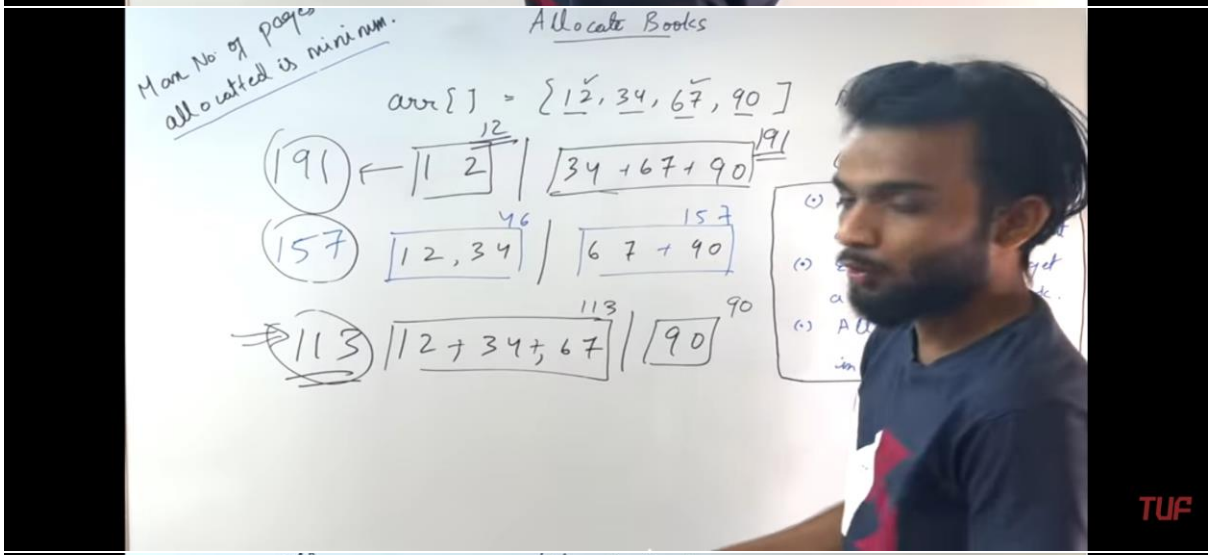
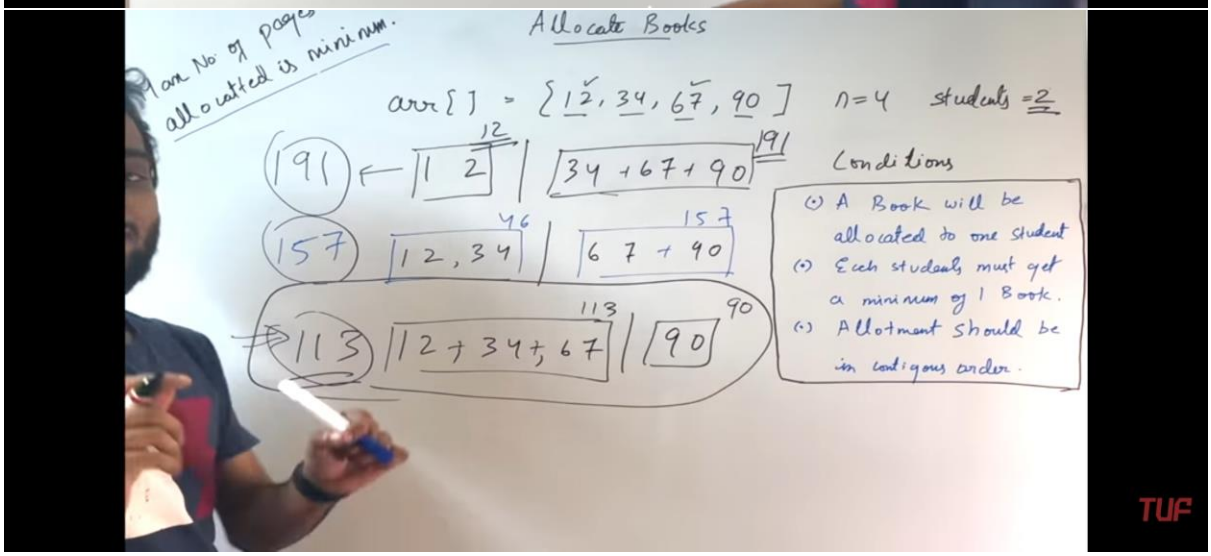

Allocate Books
arr[] = {12, 34, 67, 90} n=4 students=2
Conditions
(·) A Book will be allocated to one student
(·) Each students must get a minimum of 1 Book.
(·) Allotment should be in contigous order.
TUF

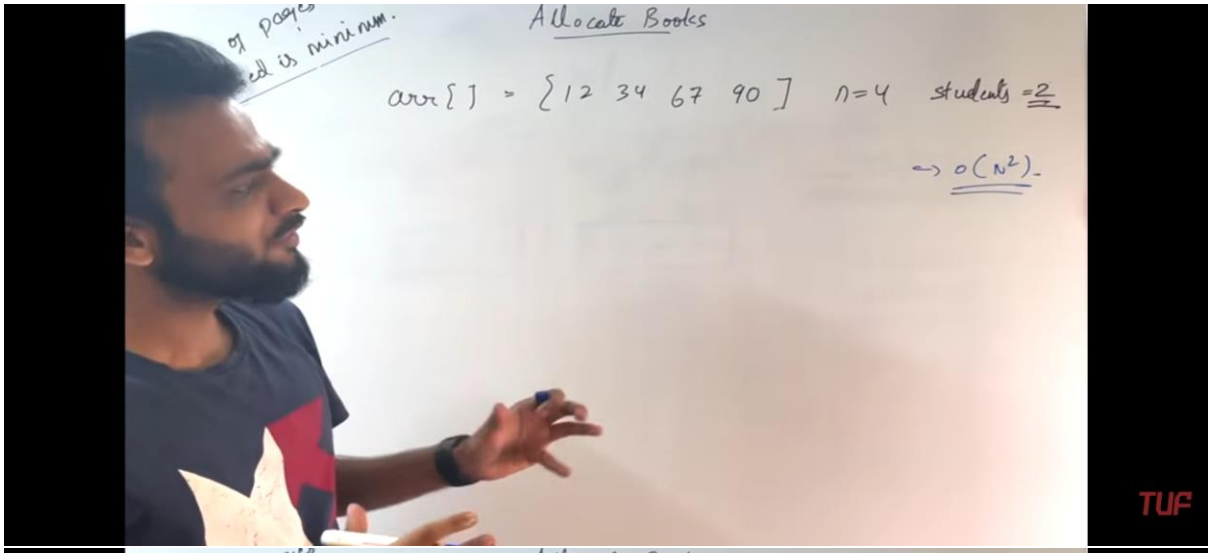

Allocate Books
arr[] = {12, 34, 67, 90} n=4 students=2
12 67 | 34 90
1 2
Conditions
(·) A Book will be allocated to one student
(·) Each students must get a minimum of 1 Book.
(·) Allotment should be in contigous order.
TUF

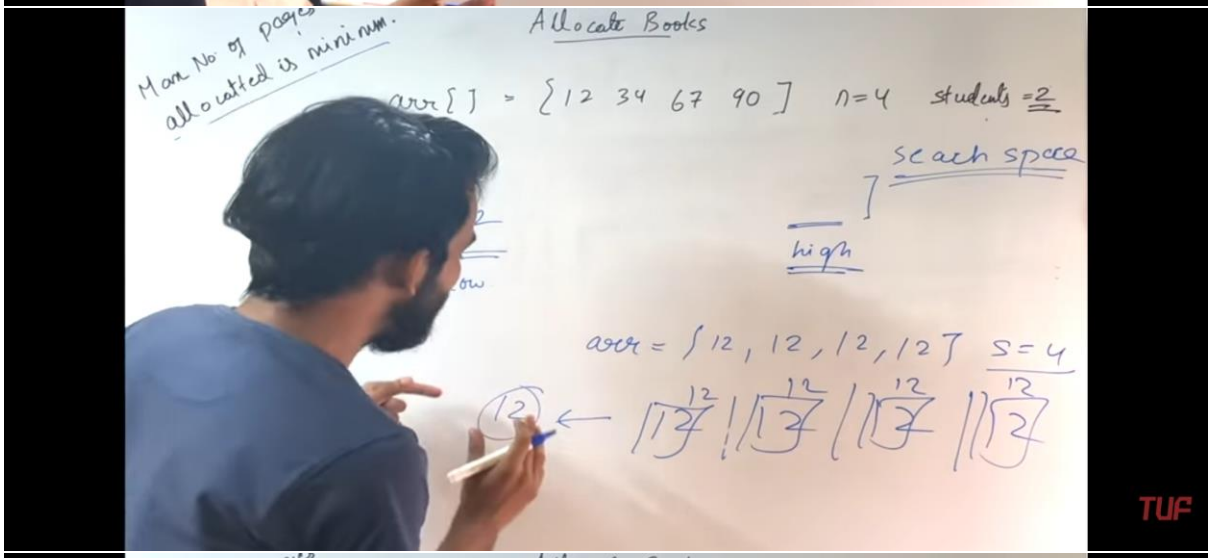

Allocate Books
arr[] = {12, 34, 67, 90} n=4 students=2
1 2 | 34 67 90
12, 34 | 67 90
12, 34, 67 | 90
Conditions
(·) A Book will be allocated to one student
(·) Each students must get a minimum of 1 Book.
(·) Allotment should be in contigous order.
TUF

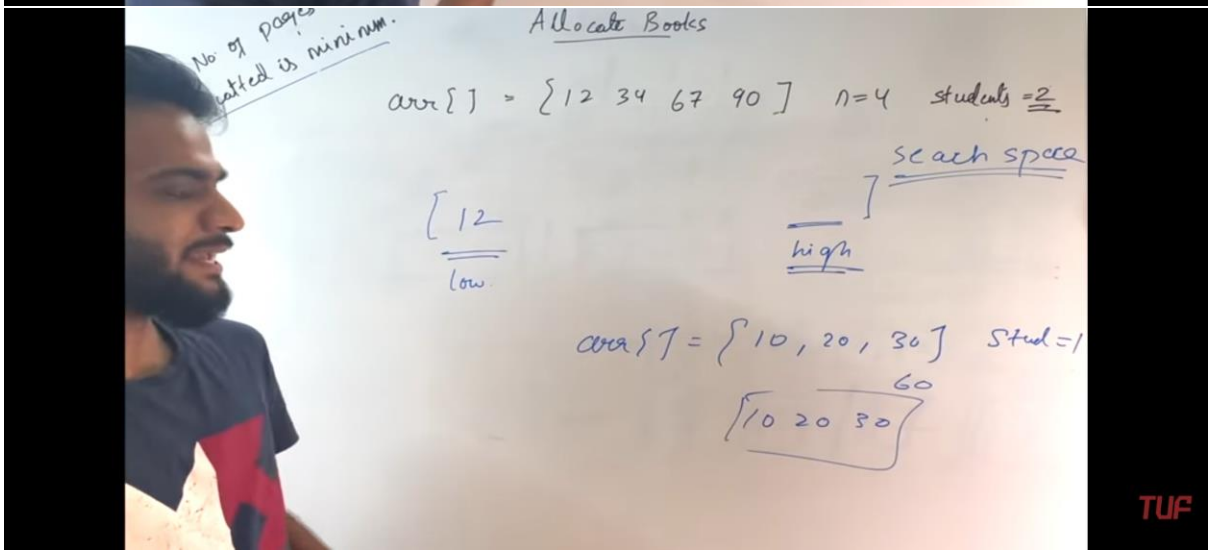

Allocate Books
Max No of pages allocated is minimum.
arr[] = {12, 34, 67, 90}
n=4 students=2
Conditions
(·) A Book will be allocated to one student
(·) Each students must get a minimum of 1 Book.
(·) Allotment should be in contigous order.
191
157
113
12 + 34 + 67
90
TUF

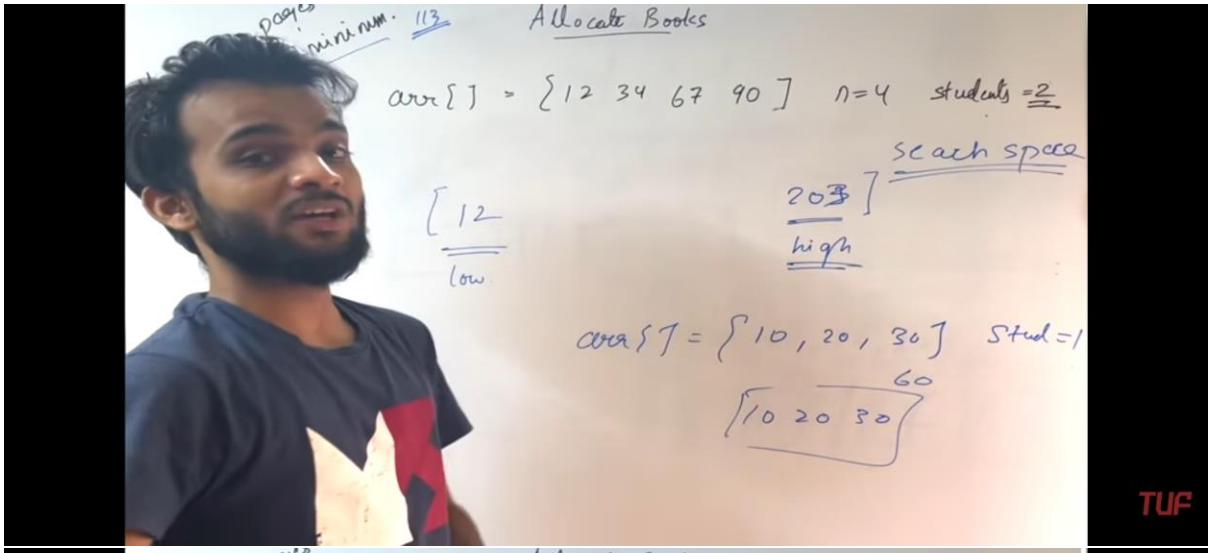

Allocate Books
Max No of pages allocated is minimum.
arr[] = {12, 34, 67, 90}
191
12
34 + 67 + 90
157
46
12, 34
67 + 90
113
12 + 34 + 67
90
TUF

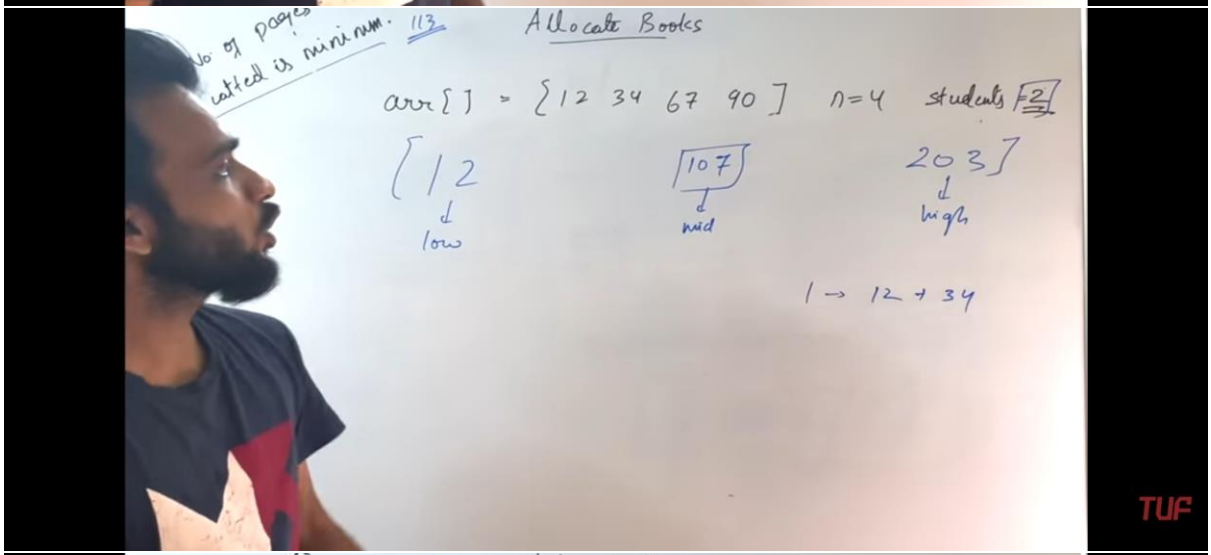

Allocate Books
Max No of pages allocated is minimum.
arr[] = {12, 34, 67, 90}
n=4 students=2
Conditions
(·) A Book will be allocated to one student
(·) Each students must get a minimum of 1 Book.
(·) Allotment should be in contigous order.
191
12
34 + 67 + 90
157
46
12, 34
67 + 90
113
12 + 34 + 67
90
TUF

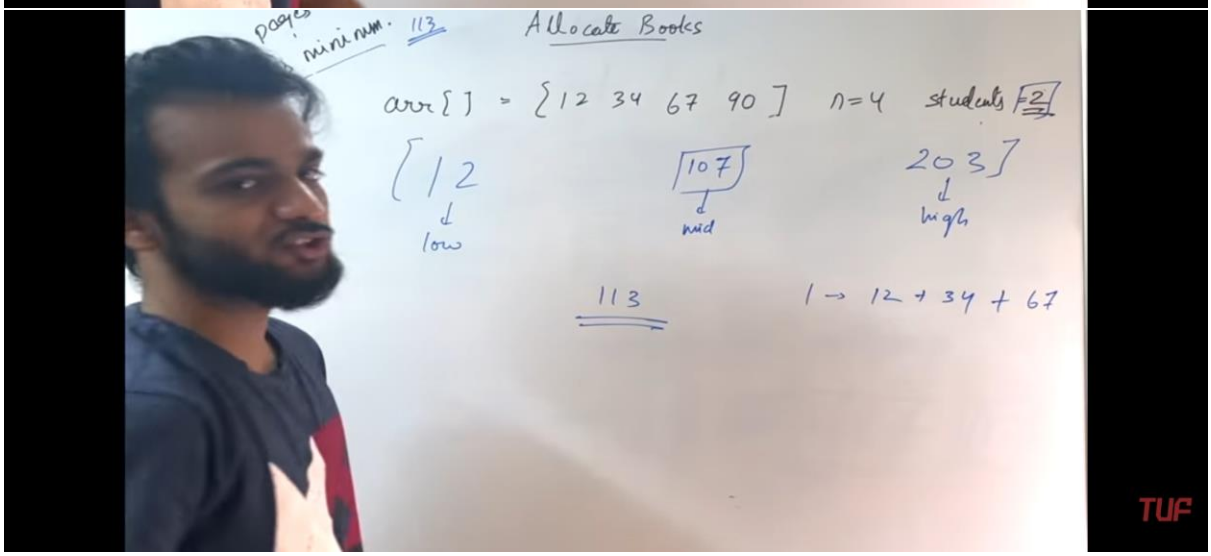

of pages
ed is minimum.
Allocate Books
arr[ ] = { 12 34 67 90 } n=4 students = 2
→ O(N²).
TUF

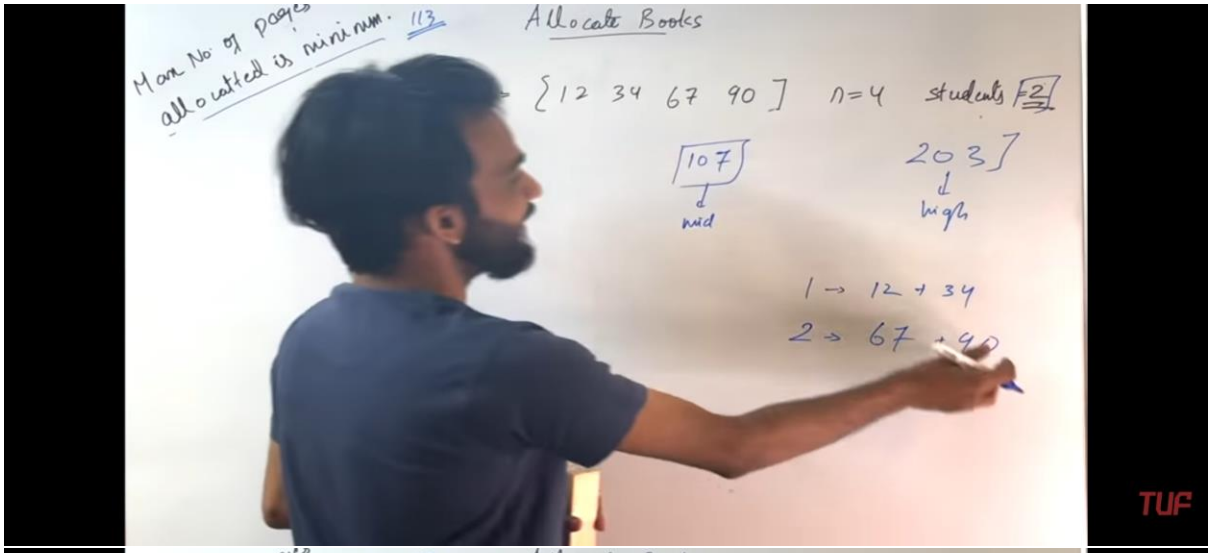

Max No of pages
allocated is minimum.
Allocate Books
arr[ ] = { 12 34 67 90 } n=4 students = 2
search space
high
low
arr = { 12, 12, 12, 12 } S = 4
12
TUF

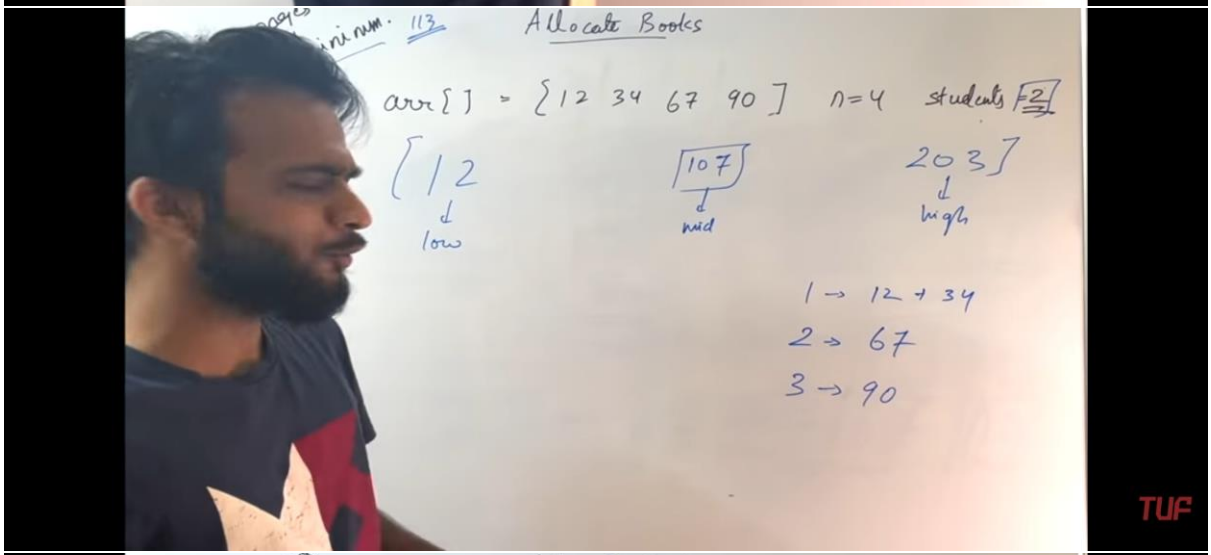

No of pages
allocated is minimum.
Allocate Books
arr[ ] = { 12 34 67 90 } n=4 students = 2
search space
[12
low
high
arr[ ] = { 10, 20, 30 } Stud = 1
60
10 20 30
TUF

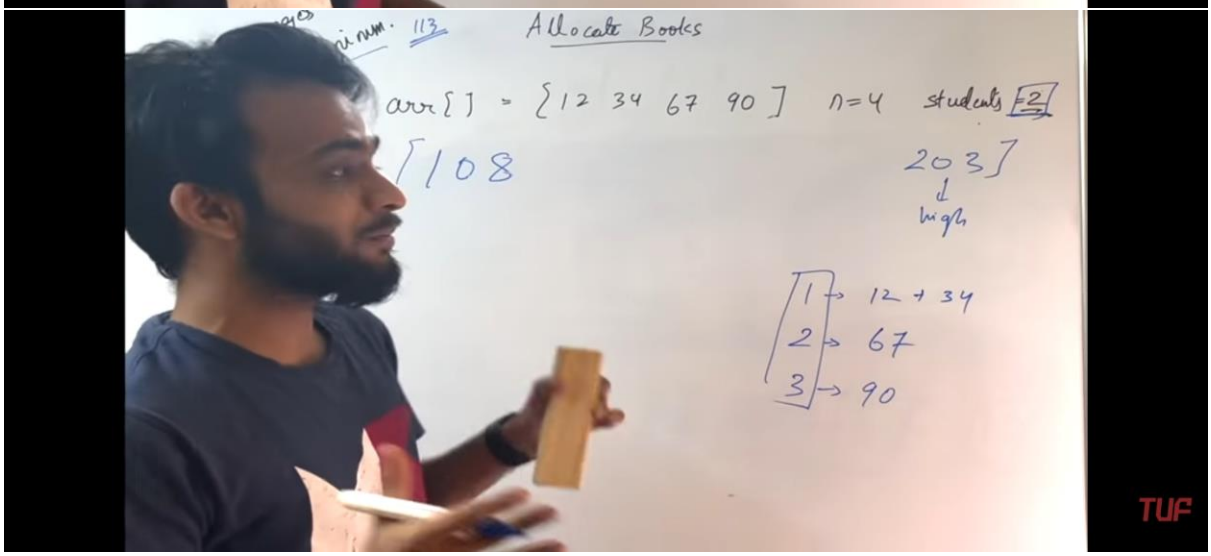

pages
minimum. 113
Allocate Books
arr[] = {12 34 67 90} n=4 students=2
search space
[12 203]
low high
arr[] = {10, 20, 30} Stud=1
60
[10 20 30]
TUF

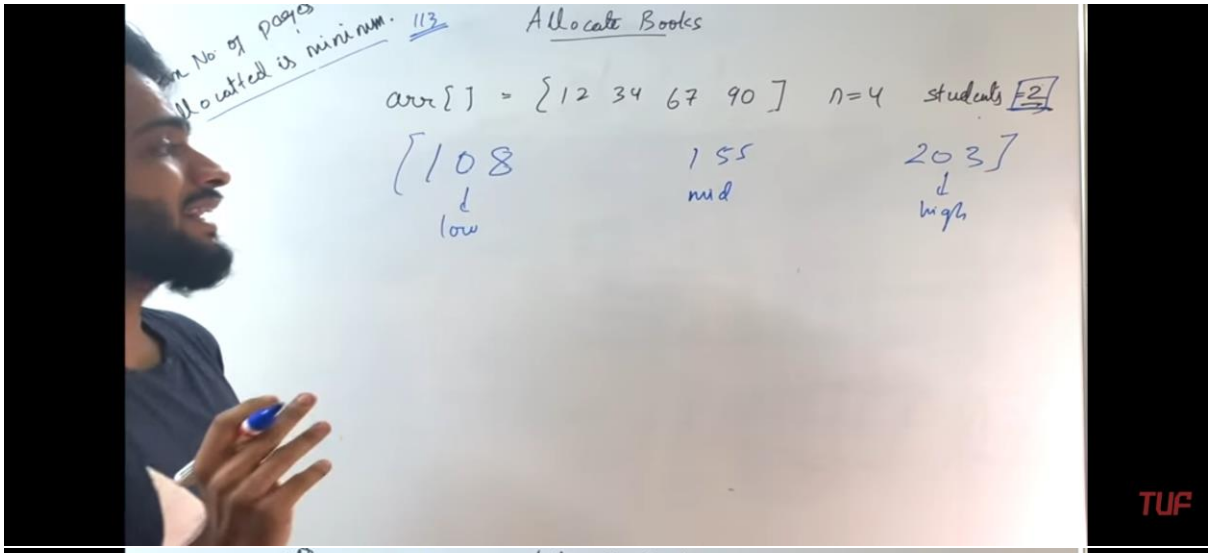

No. of pages
allotted is minimum. 113
Allocate Books
arr[] = {12 34 67 90} n=4 students=2
[12 107 203]
low mid high
1 → 12 + 34
TUF

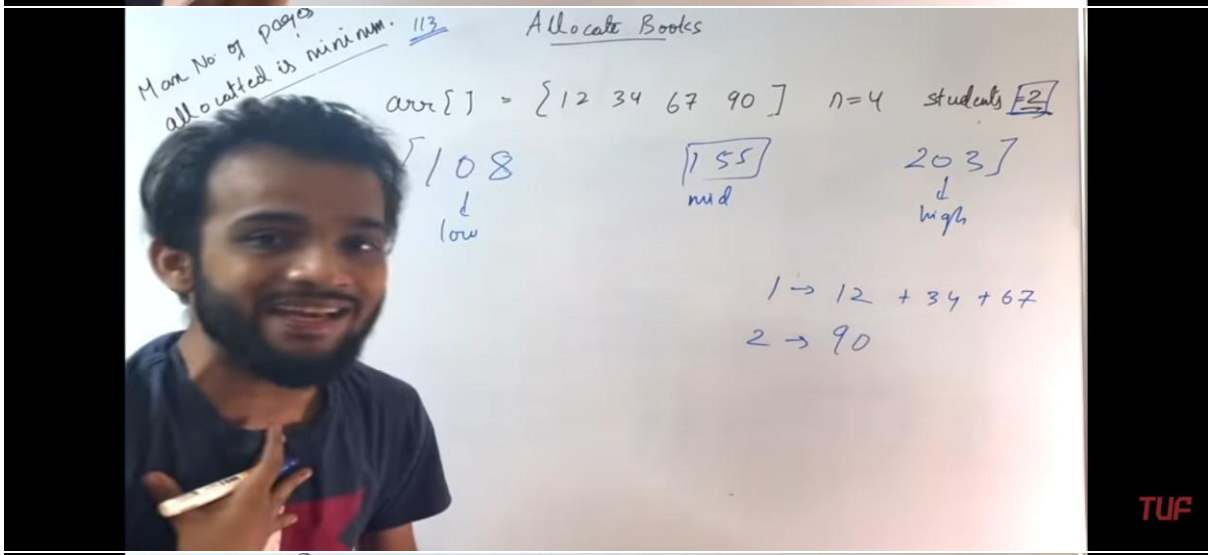

pages
minimum. 113
Allocate Books
arr[] = {12 34 67 90} n=4 students=2
[12 107 203]
low mid high
113
1 → 12 + 34 + 67
TUF

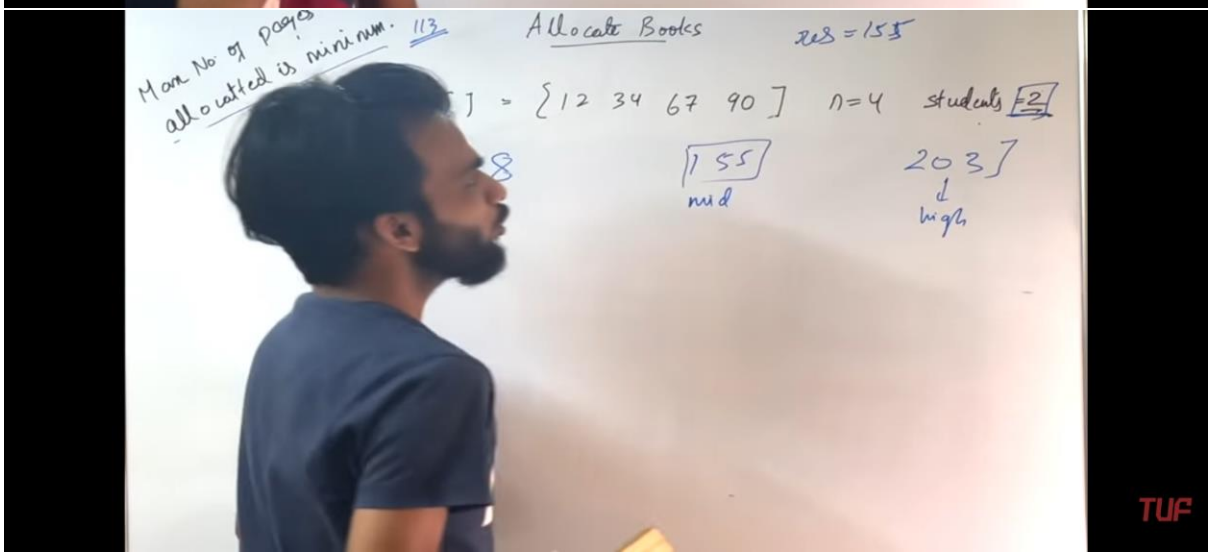

Allocate Books
arr[] = {12 34 67 90} n=4 students=2
107
mid
203
high
1 → 12 + 34
2 → 67 + 90
TUF

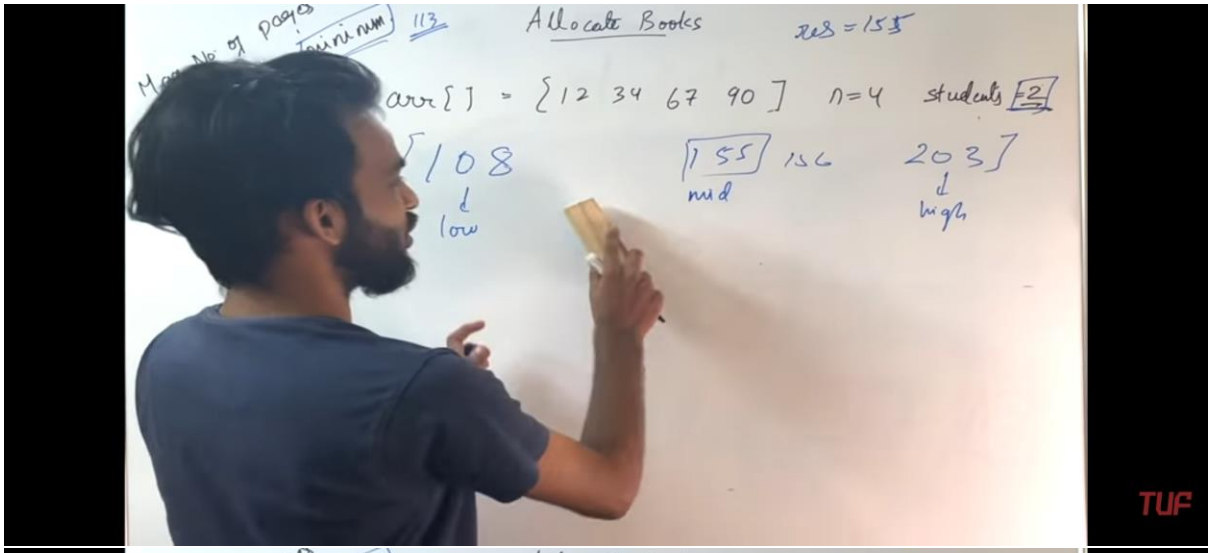

Allocate Books
arr[] = {12 34 67 90} n=4 students=2
12
low
107
mid
203
high
1 → 12 + 34
2 → 67
3 → 90
TUF

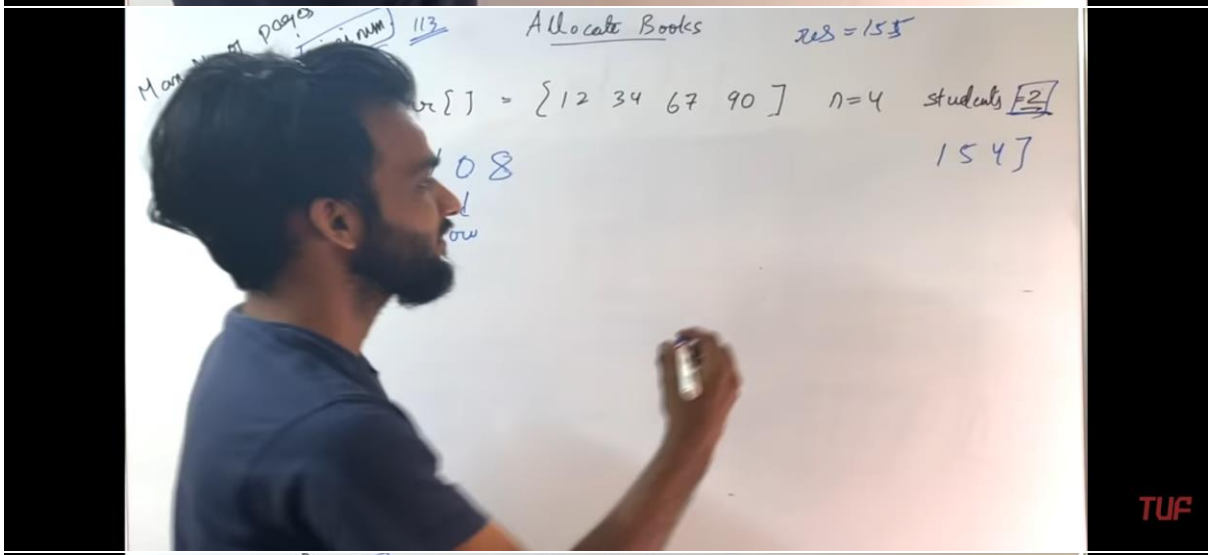

Allocate Books
arr[] = {12 34 67 90} n=4 students=2
108
203
high
1 → 12 + 34
2 → 67
3 → 90
TUF

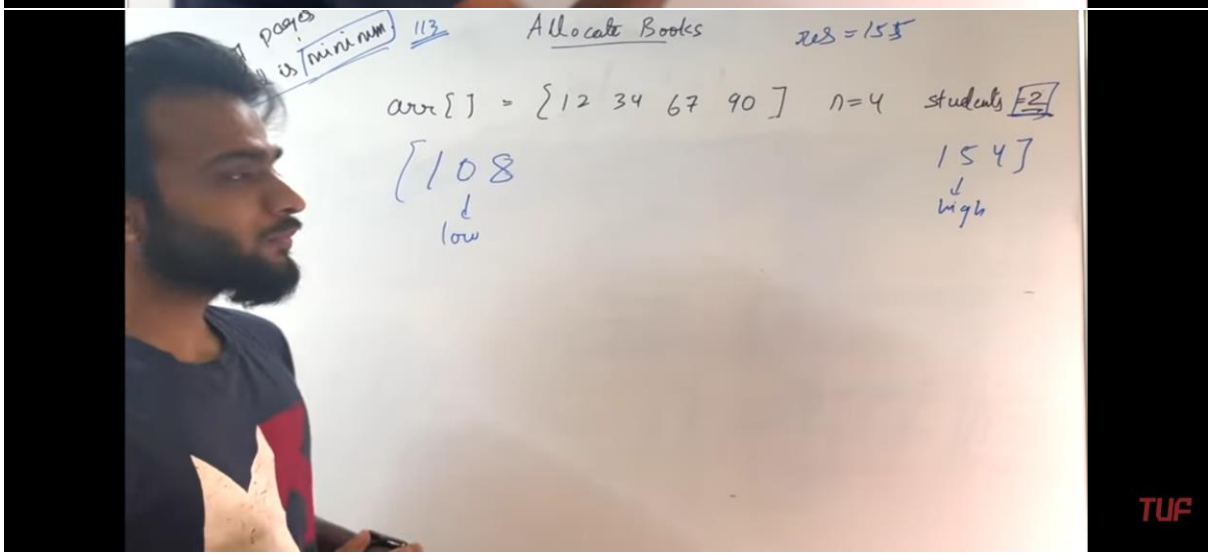

Allocate Books
arr[] = {12 34 67 90} n=4 students=2
[108 155 203]
low mid high
TUF

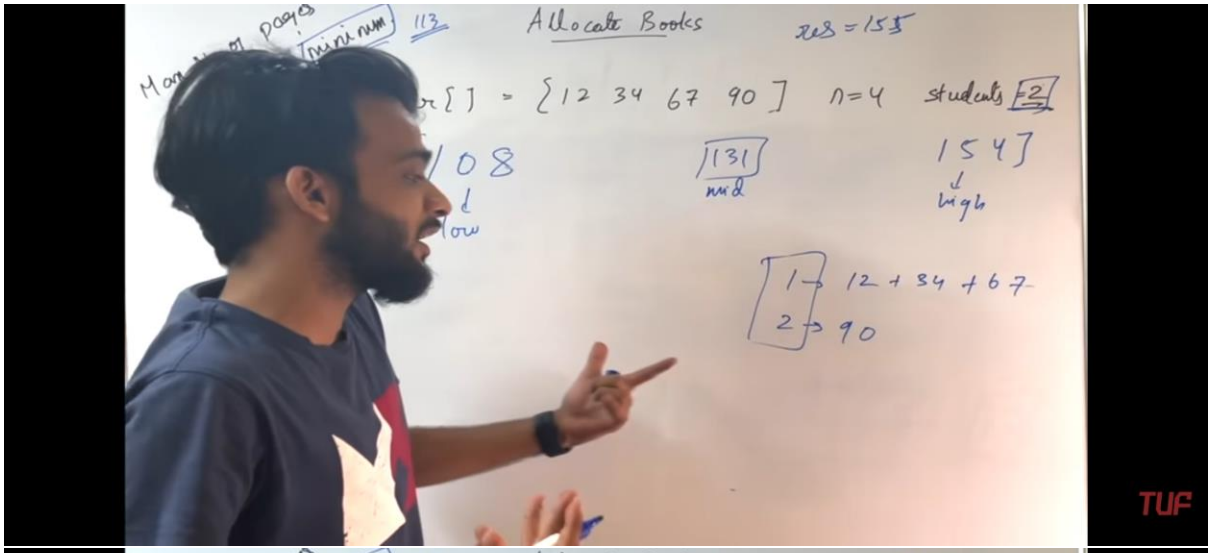

Allocate Books
arr[] = {12 34 67 90} n=4 students=2
[108 155 203]
low mid high
1 → 12 + 34 + 67
2 → 90
TUF

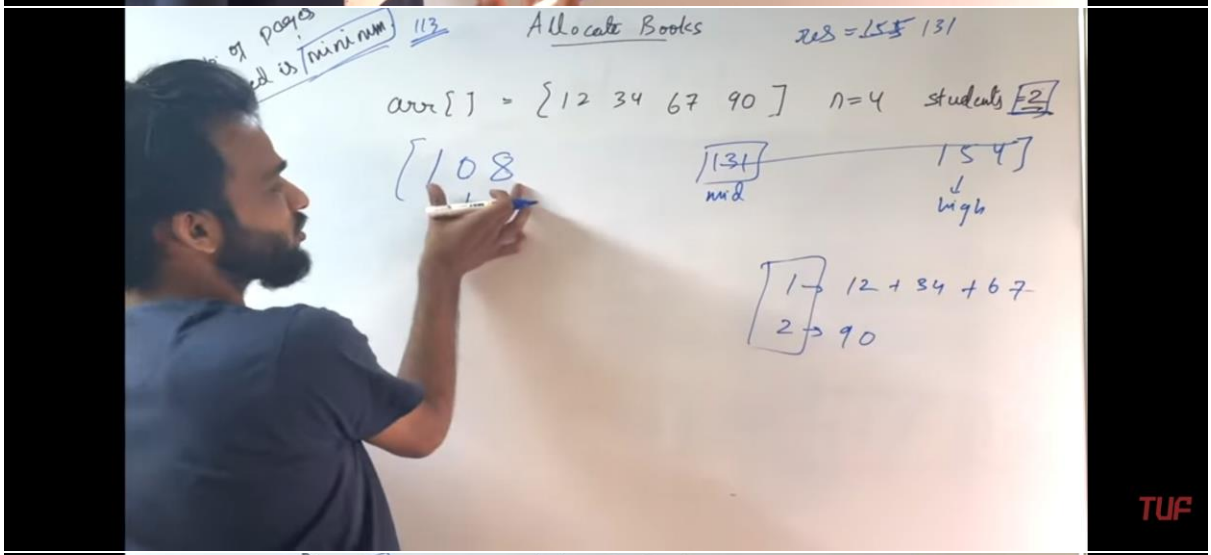

Allocate Books
ans = 155
n=4 students=2
155
mid
203
high
TUF

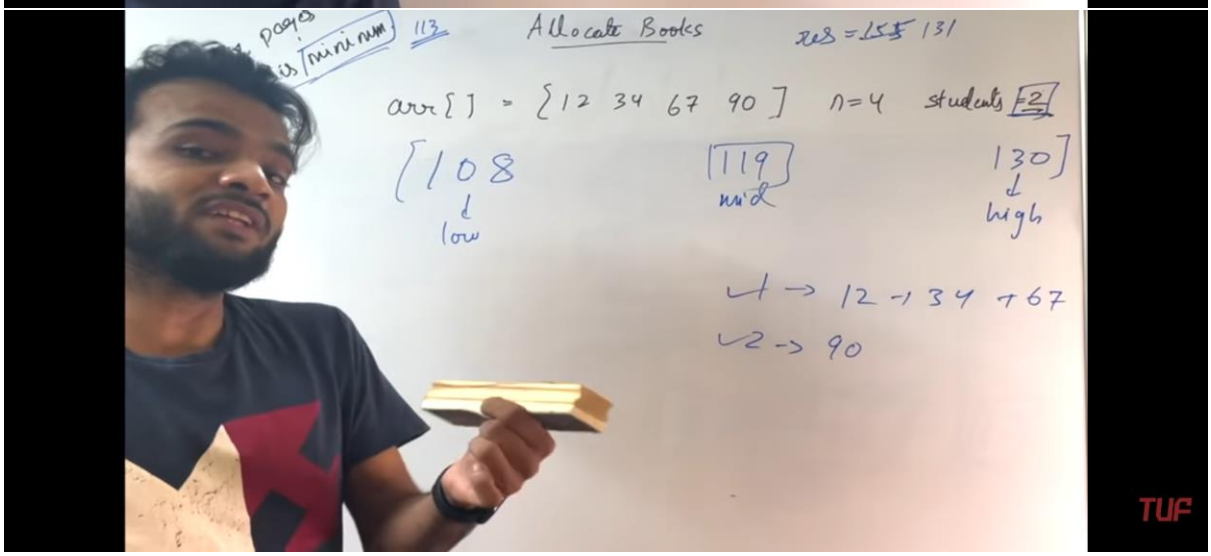

Allocate Books
res = 155
arr[] = {12 34 67 90} n=4 students F2
[108 155 156 203]
low mid high
TUF

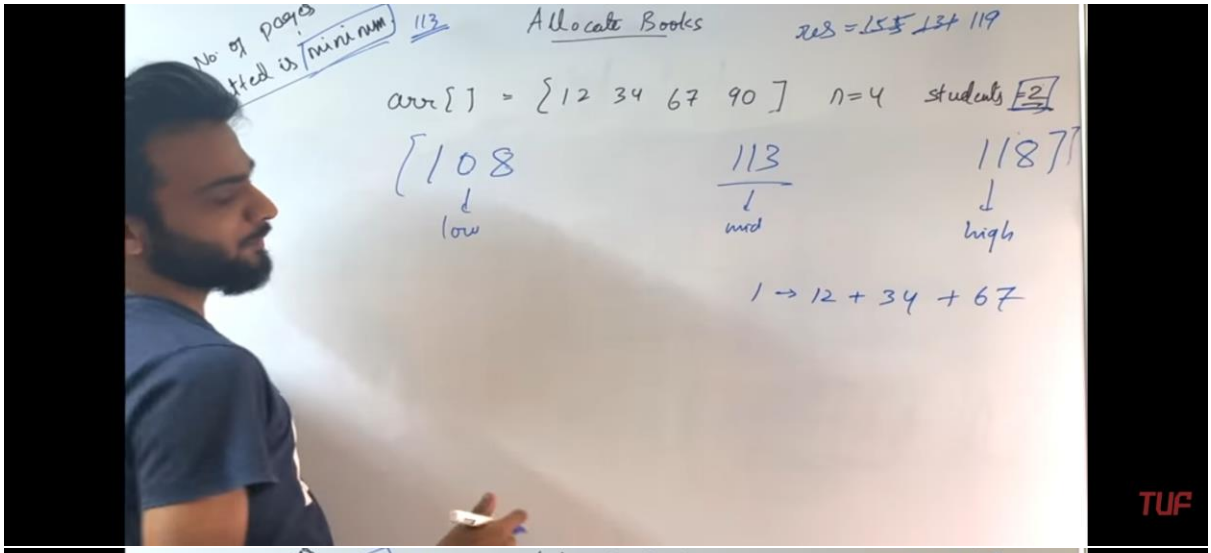

Allocate Books
res = 155
arr[] = {12 34 67 90} n=4 students F2
108 154]
low
TUF

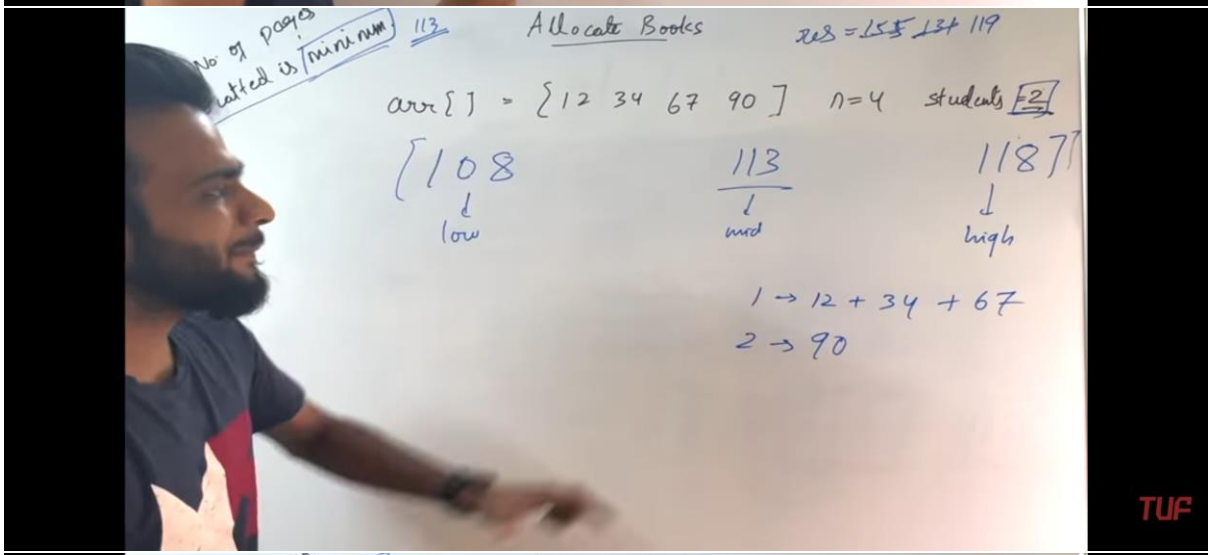

113
Allocate Books
res = 155
arr[] = {12 34 67 90} n=4 students F2
[108 154]
low high
TUF

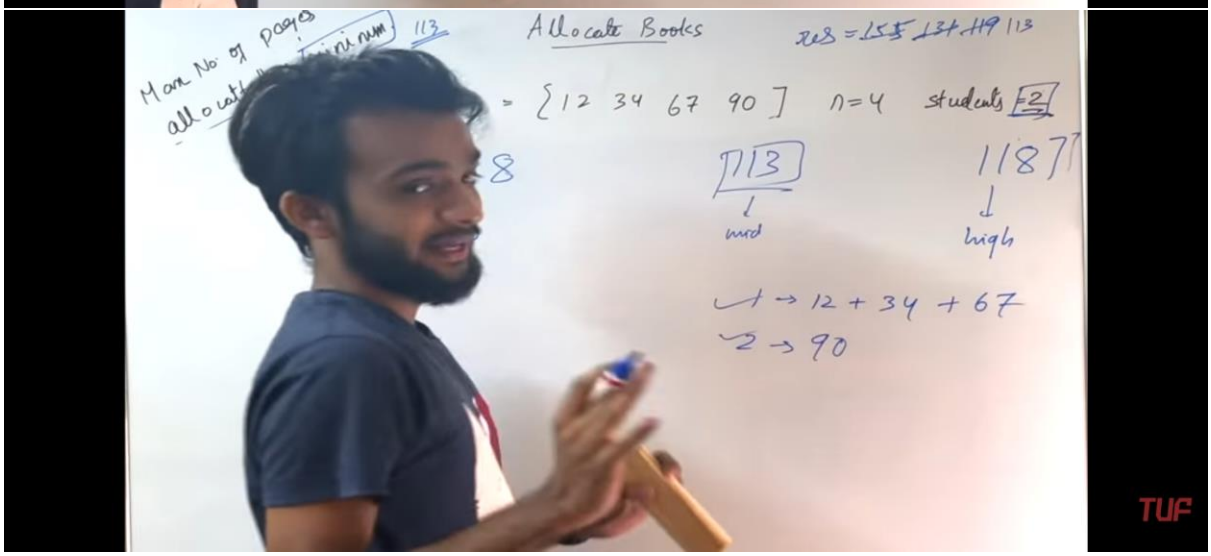

Allocate Books
res = 154
arr[] = {12 34 67 90} n=4 students F2
108
low
131
mid
154
high
1 → 12 + 34 + 67
2 → 90
TUF

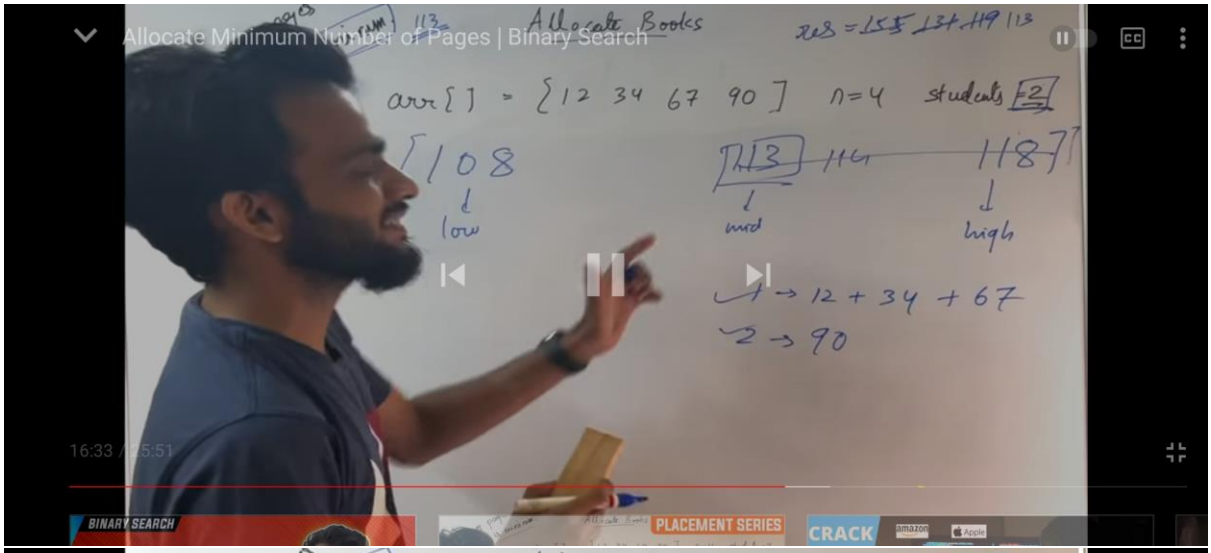

Allocate Books
res = 154 131
arr[] = {12 34 67 90} n=4 students F2
108
131
mid
154
high
1 → 12 + 34 + 67
2 → 90
TUF

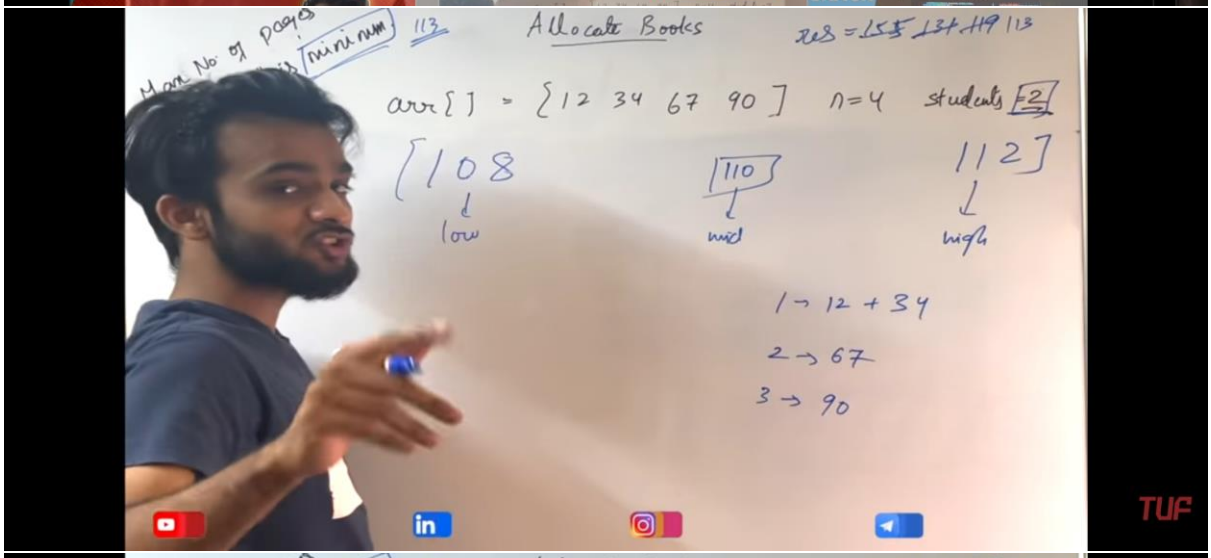

Allocate Books
res = 154 131
arr[] = {12 34 67 90} n=4 students F2
108
low
119
mid
130
high
1 → 12 + 34 + 67
2 → 90
TUF

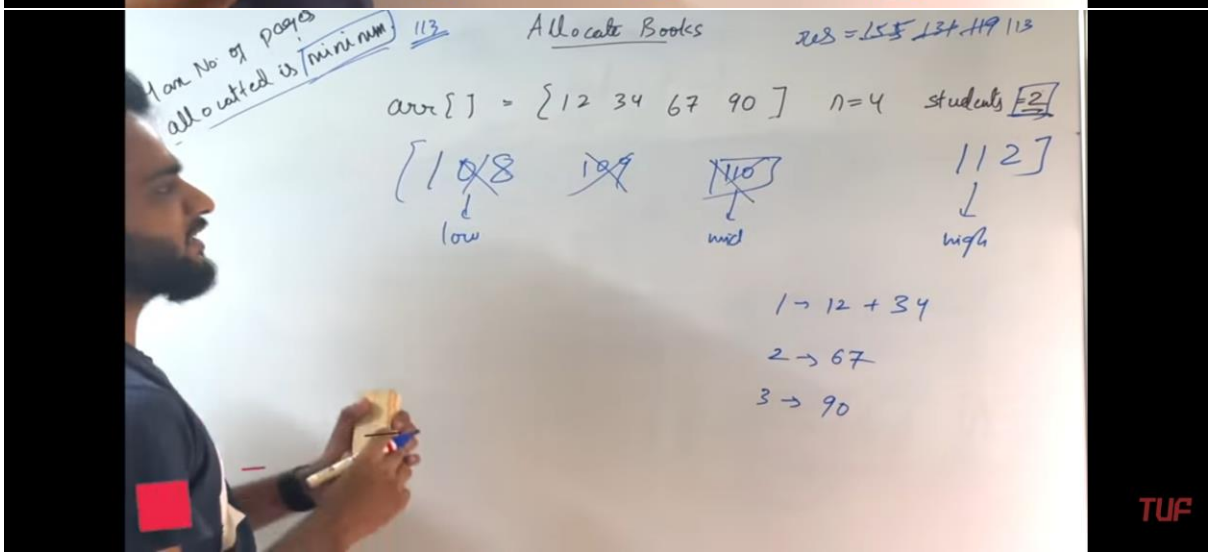

No of pages
ated is minimum
113
Allocate Books
res = 155 134 119
arr[] = {12 34 67 90} n=4 students F2
[108 113 118]
low
mid
high
1 → 12 + 34 + 67
TUF

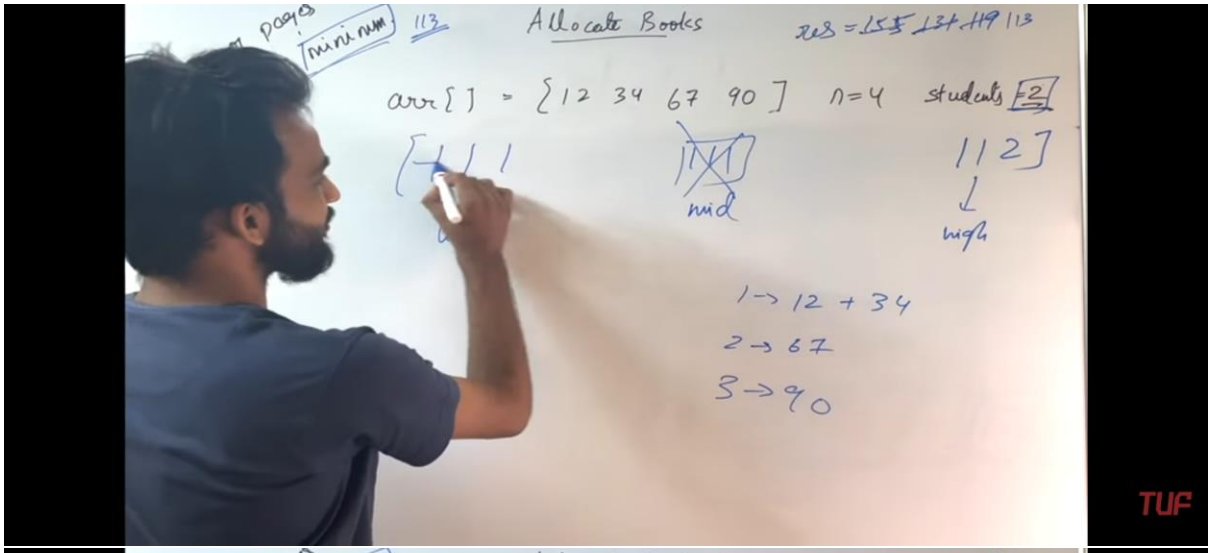

No of pages
ated is minimum
113
Allocate Books
res = 155 134 119
arr[] = {12 34 67 90} n=4 students F2
[108 113 118]
low
mid
high
1 → 12 + 34 + 67
2 → 90
TUF

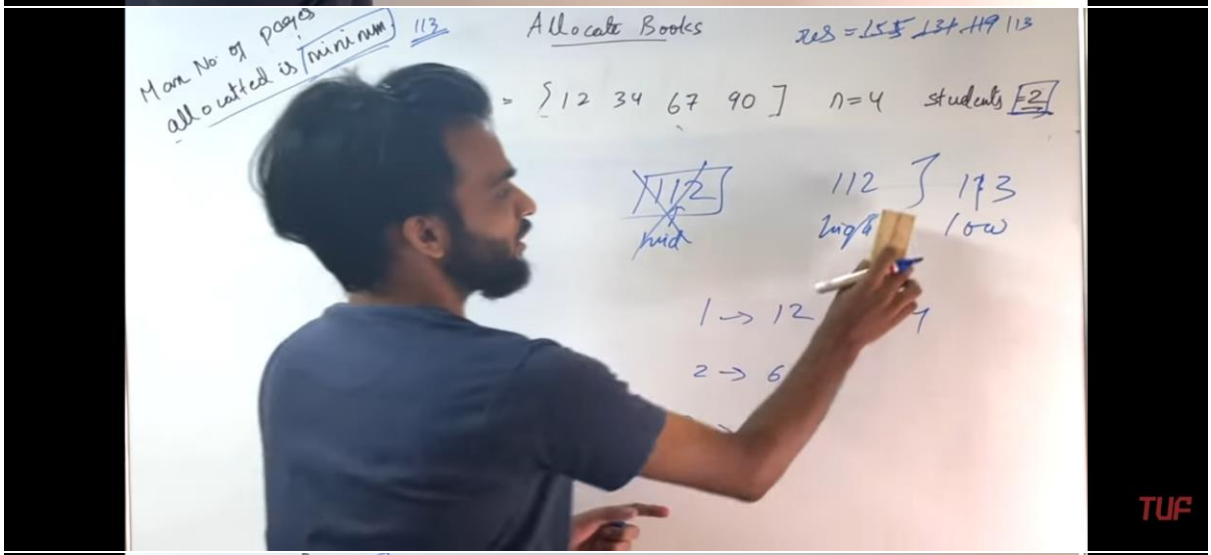

Max No of pages
allocated is minimum
113
Allocate Books
res = 155 134 119 113
= {12 34 67 90} n=4 students F2
8
113
118]
mid
high
1 → 12 + 34 + 67
2 → 90
TUF

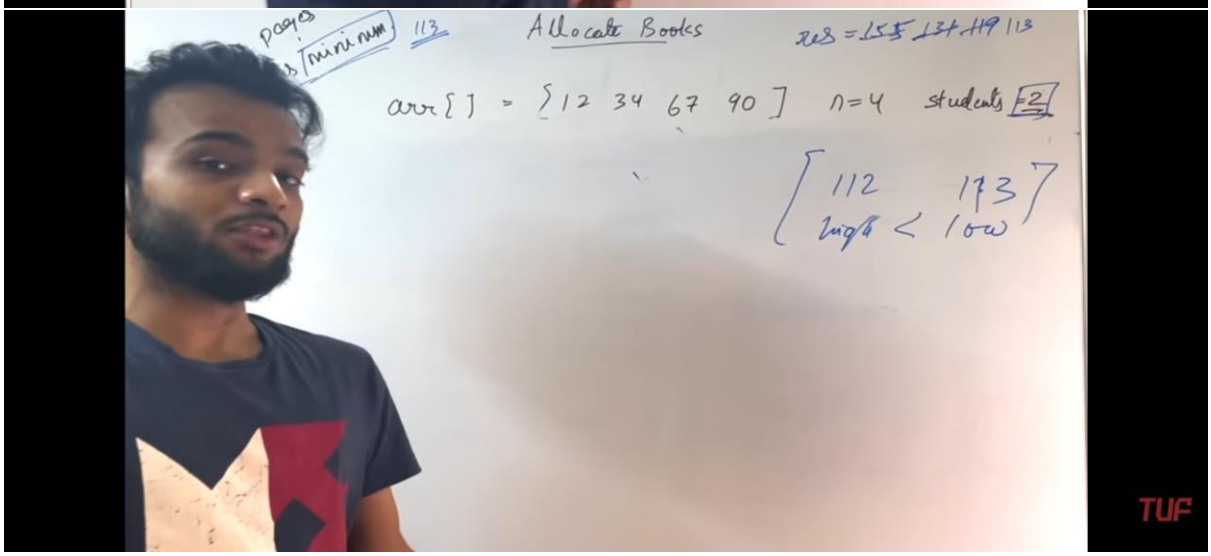

Allocate Minimum Number of Pages | Binary Search
Allocate Books
arr[] = {12 34 67 90} n=4 students=2
low
mid
high
1 → 12 + 34 + 67
2 → 90
16:33 / 25:51
BINARY SEARCH
PLACEMENT SERIES
CRACK

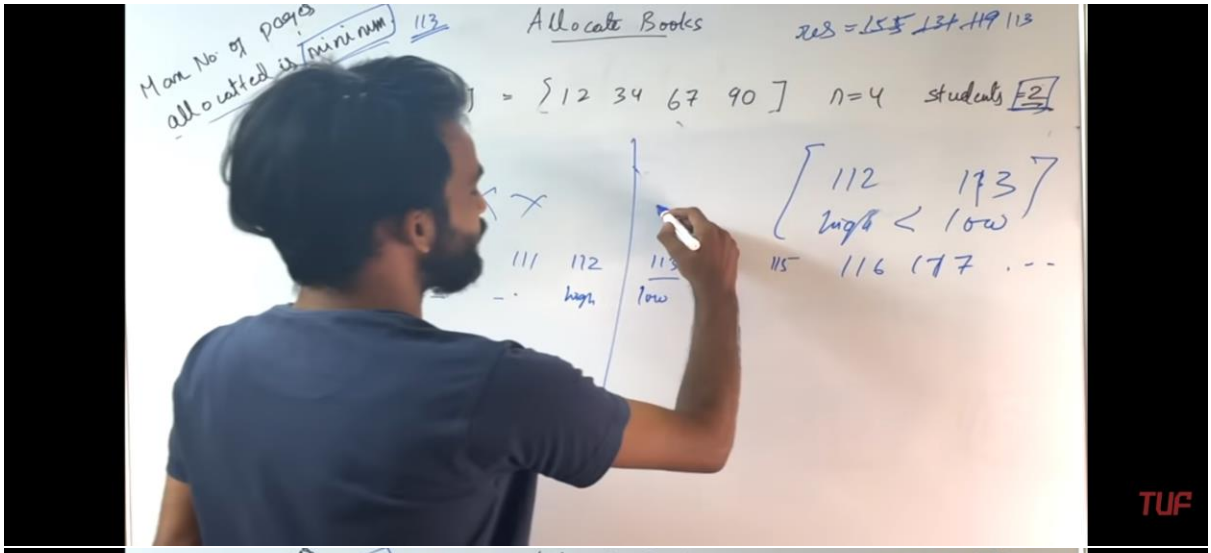

Allocate Books
arr[] = {12 34 67 90} n=4 students=2
low
mid
high
1 → 12 + 34
2 → 67
3 → 90
TUF

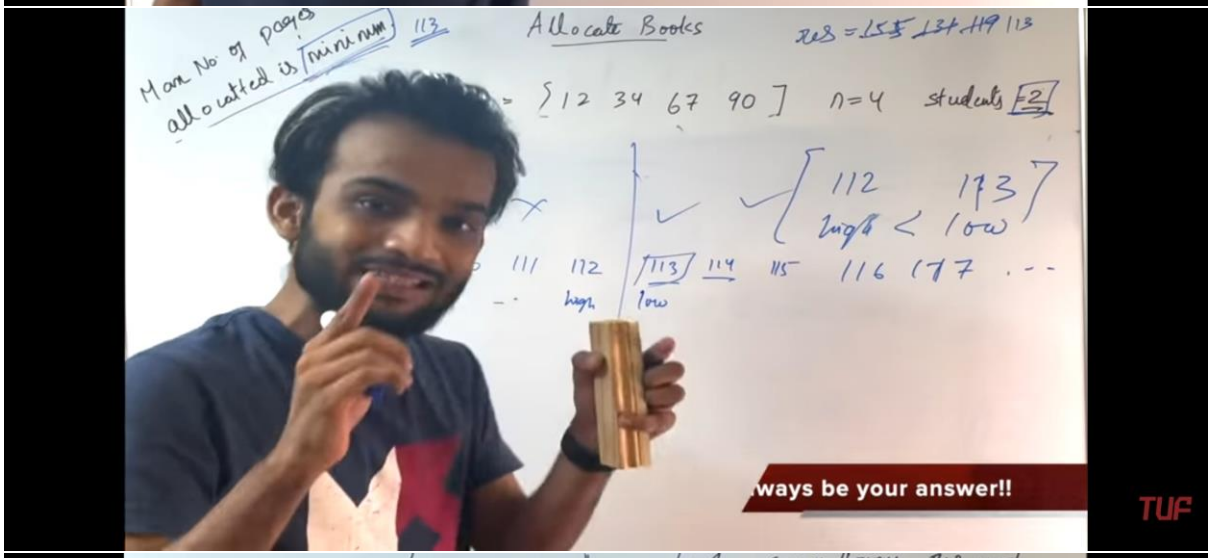

Allocate Books
arr[] = {12 34 67 90} n=4 students=2
low
mid
high
1 → 12 + 34
2 → 67
3 → 90
TUF

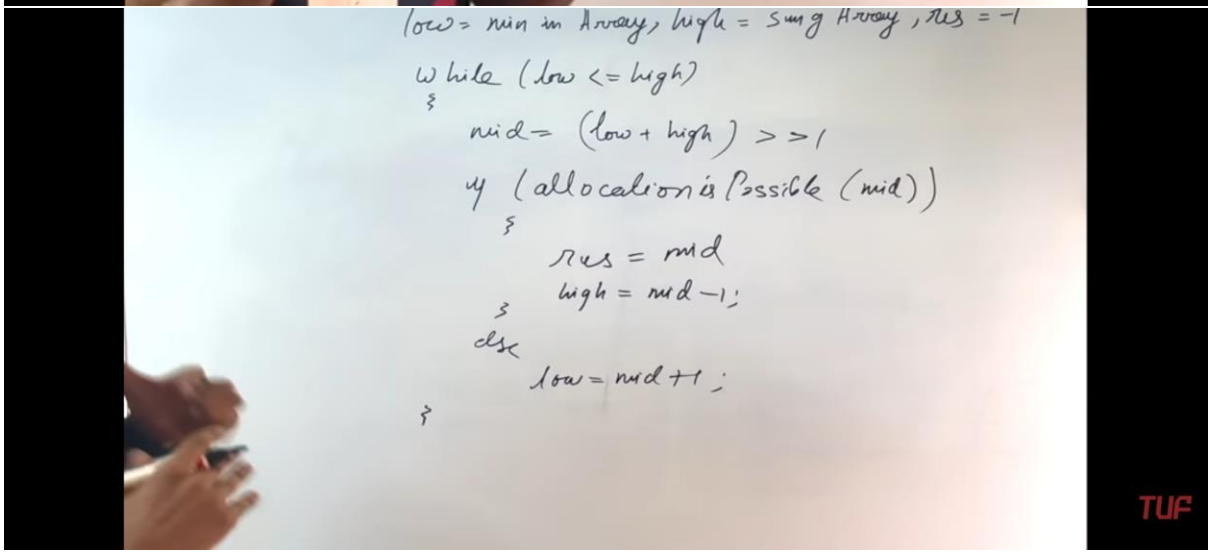

Allocate Books
arr[] = {12 34 67 90} n=4 students F2
mid
high
1 → 12 + 34
2 → 67
3 → 90
TUF


Allocate Books
mid
high
low
TUF

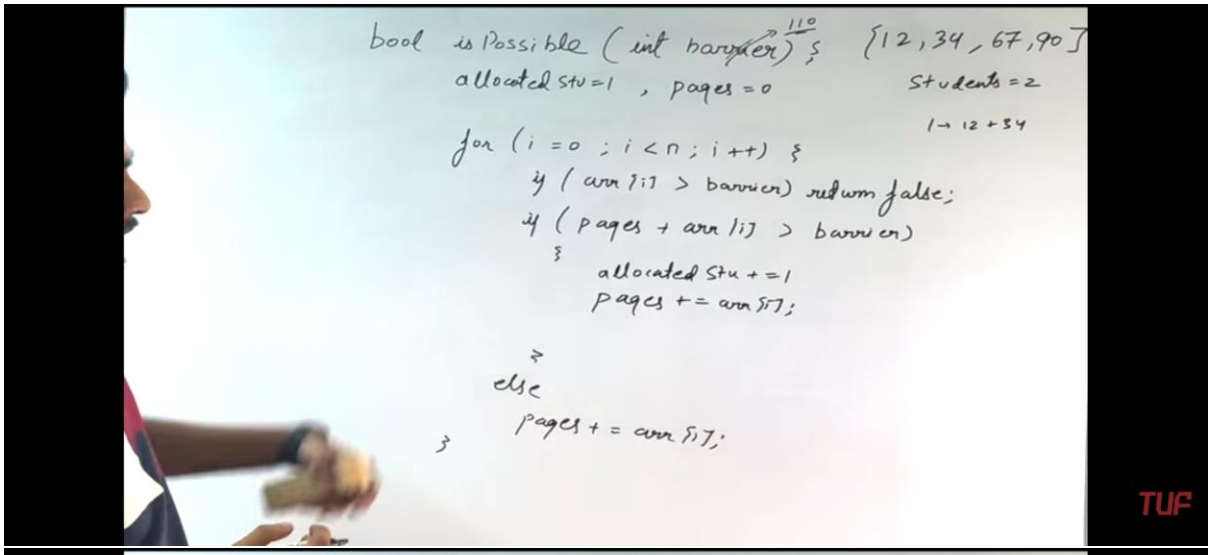

Allocate Books
arr[] = {12 34 67 90} n=4 students F2
high < low
TUF

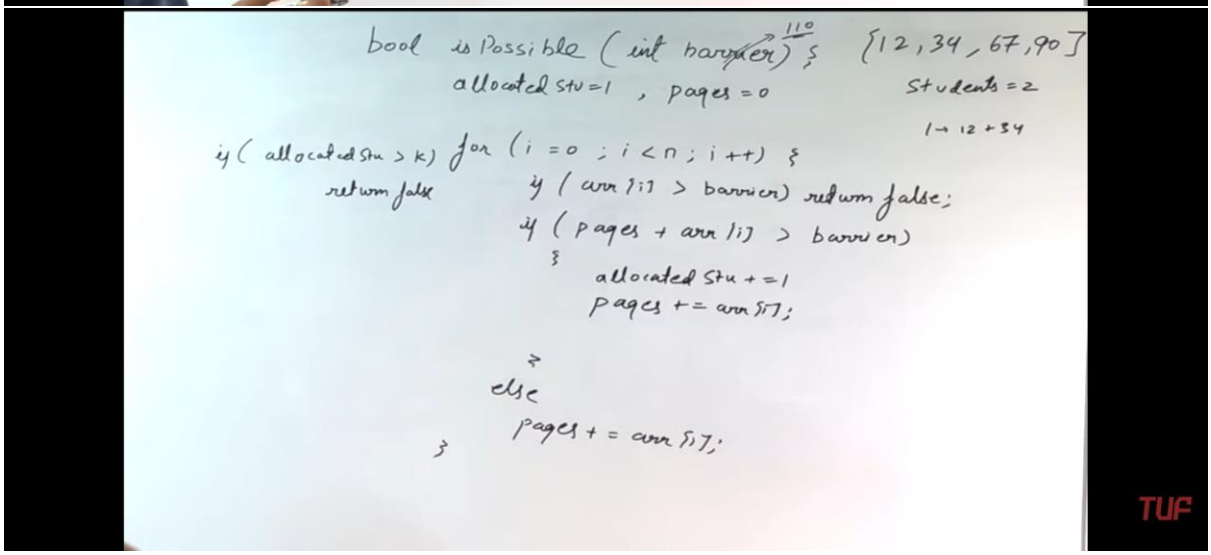


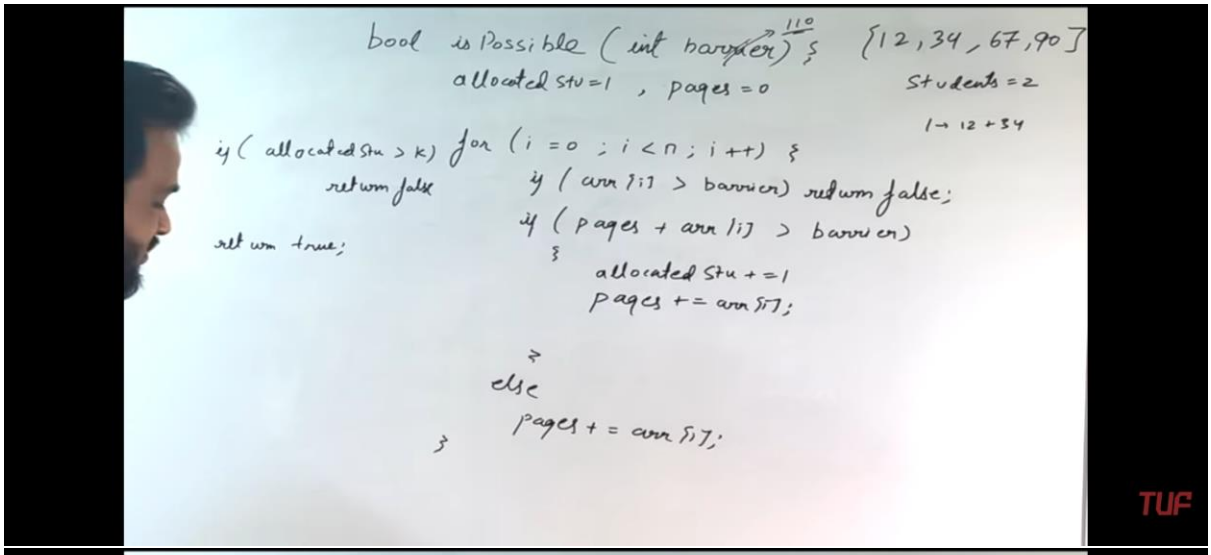


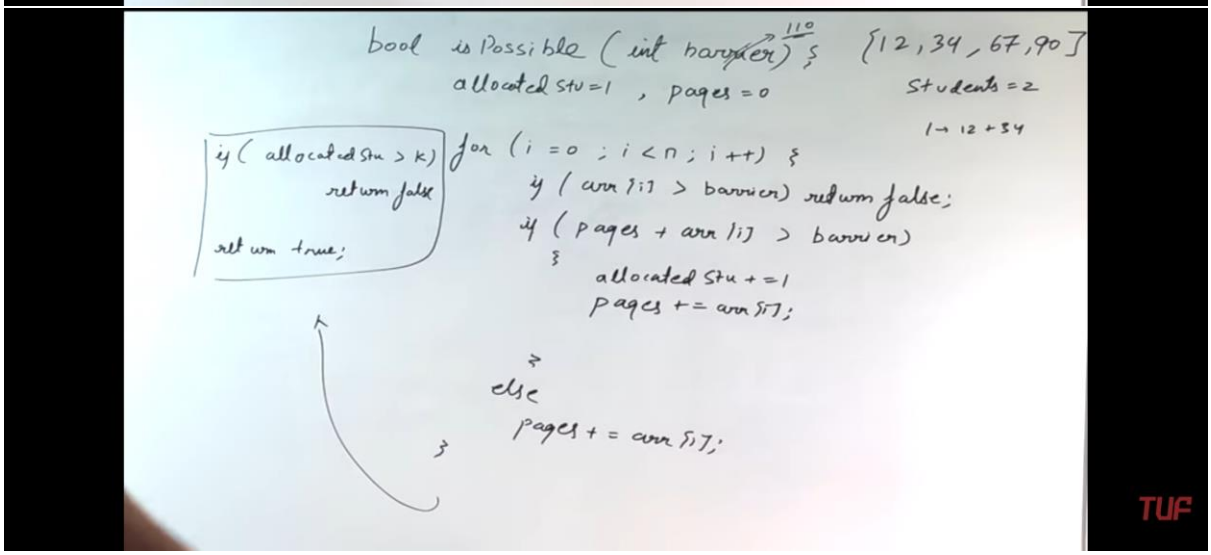


```
low = min in Array, high = sum of Array, res = -1
while (low <= high)
{
    mid = (low + high) >> 1
    if (allocation is Possible (mid))
    {
        res = mid
        high = mid - 1;
    }
    else
        low = mid + 1;
}
```


bool isPossible (int barrier) 110
allocatedStu = 1 , pages = 0
for (i = 0 ; i < n ; i++) {
if (arr[i] > barrier) return false;
if (pages + arr[i] > barrier)
{
allocatedStu += 1
pages += arr[i];
}
else
pages += arr[i];
}
{12, 34, 67, 90}
Students = 2
1 → 12 + 34
TUF


bool isPossible (int barrier) 110
allocatedStu = 1 , pages = 0
if (allocatedStu > k) return false
for (i = 0 ; i < n ; i++) {
if (arr[i] > barrier) return false;
if (pages + arr[i] > barrier)
{
allocatedStu += 1
pages += arr[i];
}
else
pages += arr[i];
}
{12, 34, 67, 90}
Students = 2
1 → 12 + 34
TUF


bool isPossible (int barrier) 110
allocatedStu = 1 , pages = 0
{12, 34, 67, 90}
Students = 2
1 → 12 + 34
if (allocatedStu > k)
return false
return true;
for (i = 0 ; i < n ; i++) {
if (arr[i] > barrier) return false;
if (pages + arr[i] > barrier)
{
allocatedStu += 1
pages += arr[i];
}
else
pages += arr[i];
}
TUF


bool isPossible (int barrier) 110
allocatedStu = 1 , pages = 0
{12, 34, 67, 90}
Students = 2
1 → 12 + 34
if (allocatedStu > k)
return false
return true;
for (i = 0 ; i < n ; i++) {
if (arr[i] > barrier) return false;
if (pages + arr[i] > barrier)
{
allocatedStu += 1
pages += arr[i];
}
else
pages += arr[i];
}
TUF