

Ques : Capacity to ship packages within D days

[Leetcode 1011]

weights = {3, 2, 2, 4, 1, 4}

sum = 16, max = 4

l0 = 4

hi = 16

days = 3



19:12 / 01:52:23



Ques : Capacity to ship packages within D days

[Leetcode 1011]

weights = { 3, 2, 2, 4, 1, 4 }

days = 3

min^m Capacity :

Capacity = 16



Ques : Capacity to ship packages within D days [Leetcode 1011]

weights = { 3, 2, 2, 4, 1, 4 }

days = 3

min^m Capacity :

lo =

hi = sum

Capacity = 16

Day 1 → 3, 2, 2, 4, 1, 4



Ques : Capacity to ship packages within D days [Leetcode 1011]

weights = { 3, 2, 2, 4, 1, 4 }

days = 3

min^m Capacity :

Capacity = 16

lo = max element of array

hi = sum of array

Day 1 → 3, 2, 2, 4, 1, 4



Ques : Capacity to ship packages within D days [Leetcode 1011]

weights = { 3, 2, 2, 4, 1, 4 }

days = 3

min^m Capacity :

Capacity = 16

lo = max element of array

hi = sum of array

Day 1 → 3, 2, 2, 4, 1, 4



Ques : Capacity to ship packages within D days

[Leetcode 1011]

weights = { 3, 2, 2, 4, 1, 4 }

days = 3

min^m Capacity :

Capacity = 16

Day 1 → 3, 2, 2, 4, 1, 4

B-S → m

4 to 16
lo hi

lo = max element of array

mi = sum of array

Kaam

no paas rakh sakte hain



Ques : Capacity to ship packages within D days

[Leetcode 1011]

weights = {3, 2, 2, 4, 1, 4, 3}

sum = 16, max = 4

days = 3

O(n) time complexity



Ques : Capacity to ship packages within D days

[Leetcode 1011]

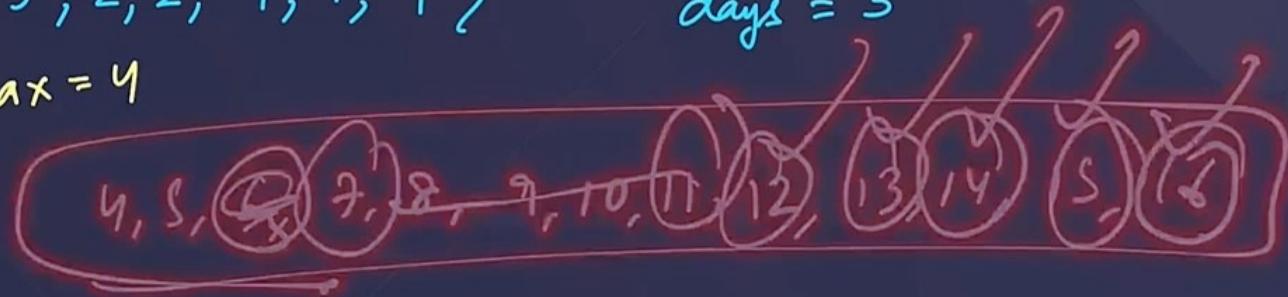
weights = {3, 2, 2, 4, 1, 4}

sum = 16, max = 4

lo = 4

hi = 16

days = 3



Ques : Capacity to ship packages within D days

[Leetcode 1011]

weights = {3, 2, 2, 4, 1, 4}

days = 3

sum = 16, max = 4

ans = 10

lo = 4

11, 1 Day - 1 : 3, 2, 2

hi = 16

Day - 2 : 4, 1, 4

mid = 10 Kaam

if (check(mid, weights, days) == true)

ans = mid;



Ques : Capacity to ship packages within D days

[Leetcode 1011]

weights = {3, 2, 2, 4, 1, 4}

days = 3

sum = 16, max = 4

ans = 10 6

lo = 4

Day-1 : 3, 2

hi = 16 9

Day-2 : 2, 4

mid = 10 6

Day-3 : 1, 4

if (check(mid, weights, days) == true)

ans = mid;

hi = mid - 1;



Ques : Capacity to ship packages within D days

[Leetcode 1011]

weights = {3, 2, 2, 4, 1, 4}

sum = 16, max = 4

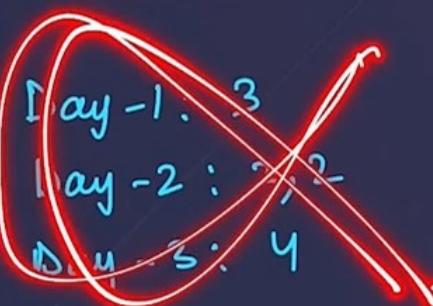
lo = 4

hi = 16 & 5

mid = 10 & 4

days = 3

ans = 10 & 6



if (check(mid, weights, days) == true)

ans = mid;

hi = mid - 1;



Ques : Capacity to ship packages within D days

[Leetcode 1011]

weights = {3, 2, 2, 4, 1, 4}

days = 3

sum = 16, max = 4

ans = 10 6

lo = 4

Day -1: 3

hi = 16 9 5

Day -2: 2, 2

mid = 10 & 4

Day -3: 4

if (check(mid, weights, days) == true) {

 ans = mid;

 hi = mid - 1;

}



Ques : Capacity to ship packages within D days

[Leetcode 1011]

weights = {3, 2, 2, 4, 1, 4}

days = 3

sum = 16, max = 4

ans = 10

lo = 4

Day -1: 3

hi = 16 & 5

Day -2: 2, 2

mid = 10 & 4

Day -3: 4

```
if (check(mid, weights, days) == true){
```

```
    ans = mid;
```

```
    hi = mid - 1;
```

```
}
```

 else

```
lo = mid + 1;
```



Ques : Capacity to ship packages within D days

[Leetcode 1011]

weights = {3, 2, 2, 4, 1, 4}

days = 3

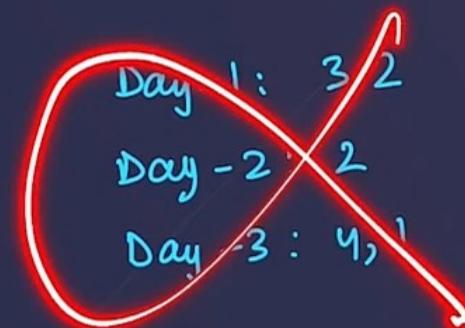
sum = 16, max = 4

ans = 10 6

lo = 1 5

hi = 16 9 5

mid = 10 & 4 5



if (check(mid, weights, days) == true) {

 | ans = mid;

 | hi = mid - 1;

} else lo = mid + 1;



Ques : Capacity to ship packages within D days

[Leetcode 1011]

weights = {3, 2, 2, 4, 1, 4}

days = 3

sum = 16, max = 4

ans = 10 6

lo \leftarrow 1 4 8 6
hi = 16 9 5

mid = 10 & 4 5

lo > hi

Day-1: 3, 2
Day-2: 2
Day-3: 4, 1

if (check(mid, weights, days) == true) {

 ans = mid;

 hi = mid - 1;



26/13 / 01:52:23 | ;



Ques : Capacity to ship packages within D days

[Leetcode 1011]

weights = {3, 2, 2, 4, 1, 4}

days = 3

sum = 16, max = 4

ans = 10 6

lo = 1 & 6

Day-1: 3, 2

hi = 16 & 5

Day-2: 2

mid = 10 & 4 & 5

Day-3: 4, 1

while ($lo \leq hi$)

if (check(mid, weights, days) == true) {

 | ans = mid;

 | hi = mid - 1;

} else lo = mid + 1;



Ques : Capacity to ship packages within D days

[Leetcode 1011]

weights = {3, 2, 2, 4, 1, 4}

days = 3

sum = 16, max = 4

ans = 10

lo = 1 & 6

Day-1: 3, 2

hi = 16 & 5

Day-2: 2

mid = 10 & 4.5

Day-3: 4, 1

```
if (check(mid, weights, days) == true){
```

```
    ans = mid;
```

```
    hi = mid - 1;
```

```
}
```

 else lo = mid + 1;



Problem List

Premium

[Description](#) [Editorial](#) [Solutions](#) [Submissions](#)

1011. Capacity To Ship Packages Within D Days

[Hint](#)

Medium



7.8K

174



Companies

A conveyor belt has packages that must be shipped from one port to another within `days` days.

The i^{th} package on the conveyor belt has a weight of `weights[i]`. Each day, we load the ship with packages on the conveyor belt (in the order given by `weights`). We may not load more weight than the maximum weight capacity of the ship.

Return the least weight capacity of the ship that will result in all the packages on the conveyor belt being shipped within `days` days.

Console ^



Run

Submit

C++ ▾ Auto

```
1 class Solution {
2 public:
3     bool check(int mid, vector<int>& weights, int days){
4         }
5     }
6     int shipWithinDays(vector<int>& weights, int days) {
7         int n = weights.size();
8         int max = INT_MIN;
9         int sum = 0;
10        for(int i=0;i<n;i++){
11            if(max<weights[i]) max = weights[i];
12            sum += weights[i];
13        }
14        int lo = max;
15        int hi = sum;
16        int minCapacity = sum;
17        while(lo<=hi){
18            int mid = lo + (hi-lo)/2;
19            if(check(mid,weights,days)){
20                minCapacity = mid;
21                hi = mid - 1;
22            }
23            else lo = mid + 1;
24        }
25    }
26 }
```



Ques : Capacity to ship packages within D days

[Leetcode 1011]

weights = {3, 2, 2, 4, 1, 4}

sum = 16, max = 4

lo = 1 & 6

hi = 16 & 5

mid = 10 & 4.5

count
= 4

Day-1: 3, 2
Day-2: 2
Day-3: 1, 1
Day-4: 4

days = 5

ans = 10.6

```
if (check(mid, weights, days) == true){
```

```
    ans = mid;
```

```
    hi = mid - 1;
```



32:19 / 01:52:23



Ques : Capacity to ship packages within D days

[Leetcode 1011]

weights = {3, 2, 2, 4, 1, 4}

sum = 16, max = 4

lo = 1 & 6

hi = 16 & 5

mid = 10 & 4.5

count = 4

Day-1: 3, 2

Day-2: 2

Day-3: 1, 1

Day-4: 4

ans = 10.6

```
if (check(mid, weights, days) == true) {
```

```
    ans = mid;
```

```
    hi = mid + 1;
```



32:20 / 01:52:23

See lo = mid + 1;





Problem List

Premium

0

[Description](#) [Editorial](#) [Solutions](#) [Submissions](#)

4th day: 9

5th day: 10

Note that the cargo must be shipped in the order given, so using a ship of capacity 14 and splitting the packages into parts like (2, 3, 4), (5), (1, 6, 7), (8), (9), (10) is not allowed.

Example 2:

Input: weights = [3,2,2,4,1,4],
days = 3

Output: 6

Explanation: A ship capacity of 6 is the minimum to ship all the packages in 3 days like this:

1st day: 3. 2

Console ^

Run

Submit

C++ Auto

```
1 class Solution {
2 public:
3     bool check(int mid, vector<int>& weights, int days){
4         // 3,2,2,4,1,4  days = 3    mid =
5         int n = weights.size();
6         int m = mid;
7         int count = 0;
8         for(int i=0;i<n;i++){
9             if(m>=weights[i]){
10                 m -= weights[i];
11             }
12             else{
13                 count++;
14                 m = mid;
15                 m -= weights[i];
16             }
17         }
18         if(count>days) return false;
19         else return true;
20     }
21     int shipWithinDays(vector<int>& weights,
22                        int n = weights.size());
23     int max = INT_MIN;
```

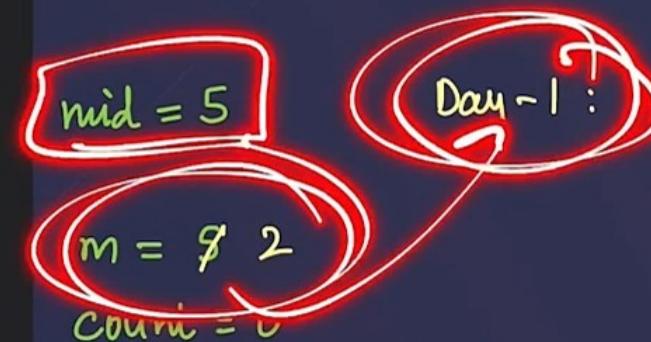


Ques : Capacity to ship packages within D days

[Leetcode 1011]

```
class Solution {
public boolean canDeliverWithinDays(int[] weights, int days) {
    // 3,2,2,4,1,4    days = 3    mid =
    int n = weights.size();
    int m = mid;
    int count = 0;
    for(int i=0;i<n;i++){
        if(m>=weights[i]){
            m -= weights[i];
        }
        else{
            count++;
            m = mid;
            m -= weights[i];
        }
    }
    if(count>days) return false;
    else return true;
}
```

0 1 2 3 4 5
3 2 2 4 1 4 $d=3$



Ques : Capacity to ship packages within D days

[Leetcode 1011]

```
// 3,2,2,4,1,4  days = 3  mid =
int n = weights.size();
int m = mid;
int count = 0;
for(int i=0;i<n;i++){
    if(m>=weights[i]){
        m -= weights[i];
    }
    else{
        count++;
        m = mid;
        m -= weights[i];
    }
}
if(count>days) return false;
else return true;
```

0 1 2 3 4 5
3 2 2 4 1 4 $d=3$

$m = 5$

$m = 5$

$count = 0$





Problem List

Premium

0

[Description](#) [Editorial](#) [Solutions](#) [Submissions](#)

4th day: 9

5th day: 10

Note that the cargo must be shipped in the order given, so using a ship of capacity 14 and splitting the packages into parts like (2, 3, 4, 5), (1, 6, 7), (8), (9), (10) is not allowed.

Testcase

Result

Accepted 4 ms

Case 1

Case 2

Case 3

Input

weights =

Console

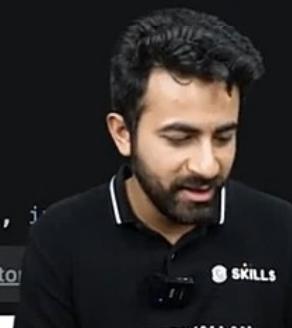


Run

Submit

C++ Auto

```
1 class Solution {
2 public:
3     bool check(int mid, vector<int>& weights, int days){
4         // 3,2,2,4,1,4  days = 3    mid =
5         int n = weights.size();
6         int m = mid;
7         int count = 1;
8         for(int i=0;i<n;i++){
9             if(m>=weights[i]){
10                 m -= weights[i];
11             }
12             else{
13                 count++;
14                 m = mid;
15                 m -= weights[i];
16             }
17         }
18         if(count>days) return false;
19         else return true;
20     }
21     int shipWithinDays(vector<int>& weights, int days) {
22         int l = 1, r = 1000000000;
```

Continue to work on your code from May 12, 2023 19:58:30 [Restore](#)

LeetCode

Problem List Premium 0

Description Editorial Solutions Submissions

Accepted

Next question

More challenges

- 410. Split Array Largest Sum
- 1231. Divide Chocolate
- 1891. Cutting Ribbons

All statuses All languages

Accepted ~ few seconds ago C++ >

Console Run Submit

Raghav Garg May 12, 2023 20:08 Details + Solution

C++

Runtime 77 ms Beats 59.74% Memory 31 MB Beats 98.93%

Click the distribution chart to view more details

Notes

Write your notes here

Related Tags

Select tags 0/5

class Solution {





Problem List

Premium

[Description](#) [Editorial](#) [Solutions](#) [Submissions](#)

1011. Capacity To Ship Packages Within D Days

Medium



7.8K

174

[Companies](#)

A conveyor belt has packages that must be shipped from one port to another within `days` days.

The i^{th} package on the conveyor belt has a weight of `weights[i]`. Each day, we load the ship with packages on the conveyor belt (in the order given by `weights`). We may not load more weight than the maximum weight capacity of the ship.

Return the least weight capacity of the ship that will result in all the packages on the conveyor belt being shipped within `days` days.

Console ^



Run

Submit

i C++ ▾

• Auto

```
22 int n = weights.size();
23 int max = INT_MIN;
24 int sum = 0;
25 for(int i=0;i<n;i++){
26     if(max<weights[i]) max = weights[i];
27     sum += weights[i];
28 }
29 int lo = max;
30 int hi = sum;
31 int minCapacity = sum;
32 while(lo<=hi){
33     int mid = lo + (hi-lo)/2;
34     if(check(mid,weights,days)){
35         minCapacity = mid;
36         hi = mid - 1;
37     } else lo = mid + 1;
38 }
39 return minCapacity;
40 } // TC = O(n*log(sum-max))
41 };
42 }
```

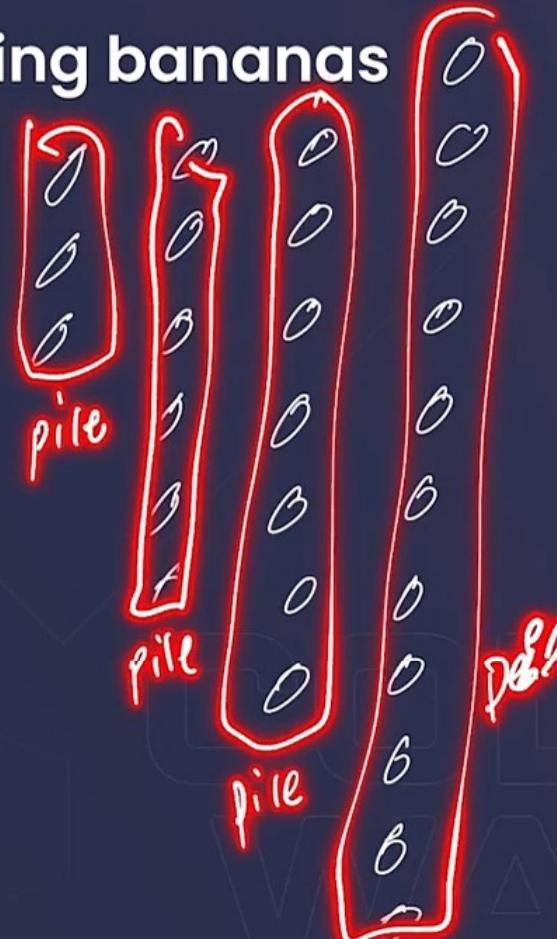
Continue to work on your code from May 12, 2023 19:58:30 [Restart](#)



Ques : Koko eating bananas

[Leetcode 875]



Ques : Koko eating bananas $n=8$ $k=3$ $\{3, 6, 7, 11\}$ 

Ques : Koko eating bananas

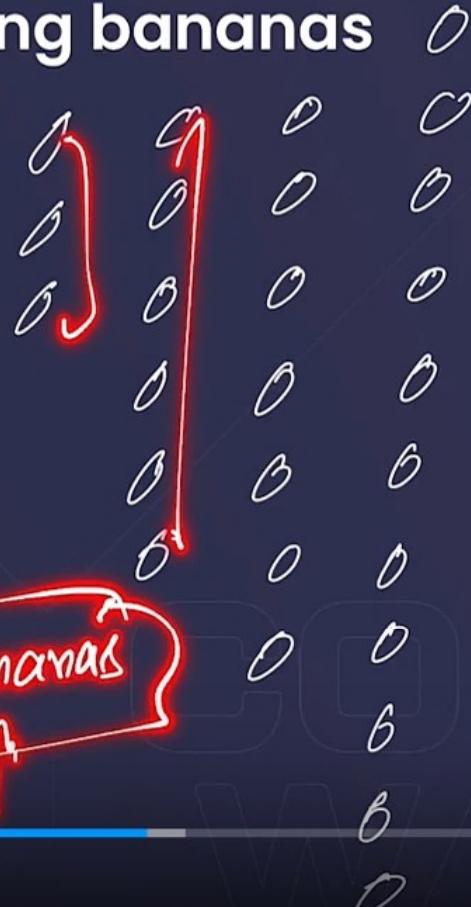
[Leetcode 875]

$n=8$

$K=3$

minimum speed

{3, 6, 7, 11}



27 bananas / hour

↓

K



Ques : Koko eating bananas

[Leetcode 875]

 $n=8$ $k=3$

minimum speed

 $[3, 6, 7, 11]$

27

2 bananas / hour

0	0	0	0
0	0	0	0
0	0	0	0
0	0	0	0
0	0	0	0
0	0	0	0
0	0	0	0
0	0	0	0



48:23 / 01:52:23



Ques : Koko eating bananas

[Leetcode 875]

 $n=8$ $k=3$

minimum speed

 $\{3, 6, 7, 11\}$

0	0	0	0
0	0	0	0
0	0	0	0
0	0	0	0
0	0	0	0
0	0	0	0
0	0	0	0
0	0	0	0

4 bananas / hour.

8 hours

3.



Ques : Koko eating bananas

[Leetcode 875]

 $n=8$ $k=3$

minimum speed

 $\{3, 6, 7, 11\}$

0	0	0	0
0	0	0	0
0	0	0	0
0	0	0	0
0	0	0	0
0	0	0	0
0	0	0	0
0	0	0	0

4 bananas / hour.

8 hours

32 bananas.



49:23

Drag from top and touch the back button to exit full screen.



Ques : Koko eating bananas

[Leetcode 875]

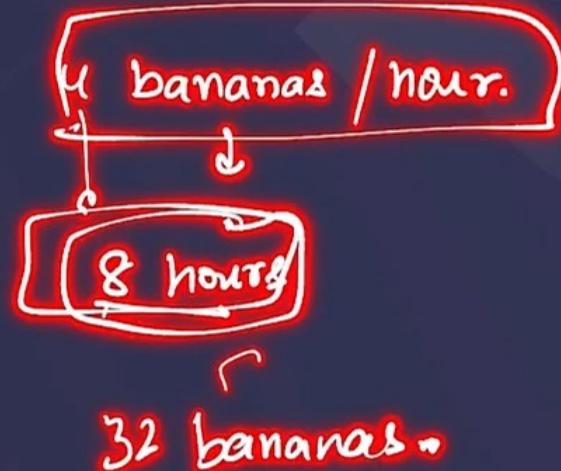
 $n=8$ $k=3$

minimum speed

 $[3, 6, 7, 11]$

$$\frac{27}{4}$$

0	0	0	0
0	0	0	0
0	0	0	0
0	0	0	0
0	0	0	0
0	0	0	0
0	0	0	0
0	0	0	0

 $K=4$ 



Ques : Koko eating bananas

[Leetcode 875]

$$h = g$$

K=3

minimum speed

$$\{3, 6, 7, 11\}$$

$$\frac{Q_f}{3} = \cancel{q} \underline{\text{hours}}$$

$$\text{Speed} = 3 \text{ bananas/hours}$$



Ques : Koko eating bananas

[Leetcode 875]

 $n=8$ $k=3$

minimum speed

 $[3, 6, 7, 11]$

count = 1

0	0	0	0
0	0	0	0
0	0	0	0
0	0	0	0
0	0	0	0
0	0	0	0
0	0	0	0
0	0	0	0

Speed = 3 bananas/hours



Ques : Koko eating bananas

[Leetcode 875]

 $n=8$ $k=3$

minimum speed

 $[3, 6, 7, 11]$

count = 12



Speed = 3 bananas/hours



51:41 / 01:52:23



Ques : Koko eating bananas

[Leetcode 875]

$n=8$

$k=3$

minimum speed

$[3, 6, 7, 11]$

count = $\lambda \geq 3$



Speed = 3 bananas/hours



51:44 / 01:52:23

Drag from top and touch the back button to exit full screen.



Ques : Koko eating bananas

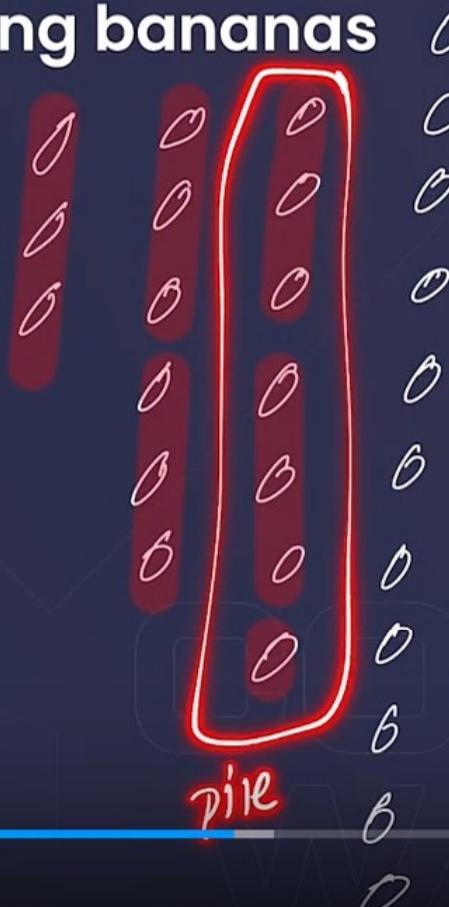
$n=8$

$k=3$

minimum speed

$[3, 6, 7, 11]$

count = $\lceil \frac{8}{3} \rceil$



[Leetcode 875]

Speed = 3 bananas/hours



52:13 / 01:52:23



Ques : Koko eating bananas $n=8$ $k=3$

minimum speed

 $[3, 6, 7, 11]$ count = $\lambda \times B \times k / 6$ 

Ques : Koko eating bananas

[Leetcode 875]

$n=8$

$k=3$

minimum speed

$[3, 6, 7, 11]$

count = 123456789
10



Speed = 3 bananas/hours



Ques : Koko eating bananas

$n=8$

$k=3$

minimum speed

$[3, 6, 7, 11]$

count = 15



[Leetcode 875]

Speed = 2 bananas / hour



Ques : Koko eating bananas

$n=8$

$k=3$

minimum speed

$[3, 6, 7, 11]$

count = 15



[Leetcode 875]

Speed = 2 bananas / hour

Binary search

↓
min speed



Ques : Koko eating bananas

[Leetcode 875]

$n=8$

$k=3$

minimum speed

$[3, 6, 7, 11]$

count = 15



Speed = 2 bananas / hour

Binary Search

↓
min speed



Ques : Koko eating bananas

[Leetcode 875]

 $n=8$ $k=3$

minimum speed

 $[3, 6, 7, 11]$

count = 15



Speed = 2 bananas / hour

Binary Search

min speed

 $lo = 1$ $hi = m > 11$ 

Ques : Koko eating bananas

[Leetcode 875]

$n=8$

$k=3$

minimum speed

$[3, 6, 7, 11]$

count = 6 hour



Speed = 6

Binary Search
↓

min speed

$lo = 1$

$hi = 11$

$mid = \frac{lo+hi}{2} = 6$



Ques : Koko eating bananas

[Leetcode 875]

$n=8$

$k=3$

minimum speed

$[3, 6, 7, 11]$

count = 6 hours

ans = 6



speed = 6

Binary Search
↓

min speed

lo = 1

hi = 11

mid = $\frac{lo+hi}{2}$



Ques : Koko eating bananas

[Leetcode 875]

$n=8$

$k=3$

minimum speed

$[3, 6, 7, 11]$

count = 6 hours

ans = 6



speed = 6

Binary Search
↓

min speed

$$lo = 1, \quad \frac{s+1}{2} = \frac{6}{2} = 3$$

$$hi = 11 - 5 = 6$$

$$mid = \frac{lo + hi}{2} = 6$$

$$hi = mid - 1$$

$$lo = mid + 1$$

Ques : Koko eating bananas

[Leetcode 875]

$n=8$

$k=3$

minimum speed

$[3, 6, 7, 11]$

count = 6 hours

ans = 6



speed = 6

Binary Search
↓

min speed

$lo = 1$

$hi = 11$

$mid = \frac{lo+hi}{2} = 3$

$hi = mid - 1$

$lo = mid + 1$

Ques : Koko eating bananas

[Leetcode 875]

 $n=8$ $k=3$

minimum speed

 $[3, 6, 7, 11]$ $\text{count} = 10$ $\text{ans} = 6$  $\text{speed} = 6$ Binary Search
↓

min speed

 $lo = 1$ $hi = 11$

$$\text{mid} = \frac{lo+hi}{2} = 3$$

 $hi = \text{mid} - 1$ $lo = \text{mid} + 1$

Ques : Koko eating bananas

[Leetcode 875]

$n=8$

$k=3$

minimum speed

$[3, 6, 7, 11]$

count = 10

ans = 6



speed = 6

Binary Search
↓

min speed

$lo = 1^4$

$hi = 11^5$

$mid = \frac{lo+hi}{2} = 3^4$

$hi = mid - 1$

$lo = mid + 1$

Ques : Koko eating bananas

[Leetcode 875]

 $n=8$ $k=3$

minimum speed

 $[3, 6, 7, 11]$

Count = 10 8

ans = 6



Speed = 6

Binary Search

↓
min speed $lo = 1^{\text{st}}$ $hi = 1^{\text{st}}$ $mid = \frac{lo + hi}{2} = 3^{\text{rd}}$ $hi = mid - 1$ $lo = mid + 1$

Ques : Koko eating bananas

[Leetcode 875]

 $n=8$ $k=3$

minimum speed

 $[3, 6, 7, 11]$

count = 10 8

ans = 6



Speed = 6

Binary Search

↓
min speed $lo = 1^{\text{st}}$ $hi = 5^{\text{th}}$ $mid = \frac{lo+hi}{2} = 3^{\text{rd}}$ $hi = mid - 1$ $lo = mid + 1$

Ques : Koko eating bananas

[Leetcode 875]

 $n=8$ $k=3$

minimum speed

 $[3, 6, 7, 11]$

count = 10 8

ans = 6 4



Speed = 6 4

Binary Search

↓
min speed $lo = 1 4$ $hi = 11 5$ $mid = \frac{lo+hi}{2} = 6 3 4$ $hi = mid - 1$ $lo = mid + 1$

Ques : Koko eating bananas

[Leetcode 875]

$n=8$

$k=3$

minimum speed

$[3, 6, 7, 11]$

count = 10 8

ans = 6 4



speed = 6 4

Binary Search

↓
min speed

$lo = 1 4$

$hi = 11 \leq 3$

$mid = \frac{lo + hi}{2} = 6 3 4$

$hi = mid - 1$

$lo = mid + 1$

Ques : Koko eating bananas**[Leetcode 875]**
$$\begin{bmatrix} 30, 11, 23, 4, 20 \end{bmatrix} \quad h=5$$
$$\begin{matrix} \downarrow & \downarrow & \downarrow & \downarrow & \downarrow \\ 2 & 1 & 2 & 1 & 2 \end{matrix} \quad k=?$$
 $lo = 1$ $hi = 30$ $mid = 15$ $count =$ 

Ques : Koko eating bananas**[Leetcode 875]**
$$\begin{array}{ccccc} [30, 11, 23, 4, 20] & h=5 \\ \downarrow & \downarrow & \downarrow & \downarrow & \downarrow \\ 2 & 1 & 2 & 1 & 2 \end{array}$$

$K=?$

 $lo = 1$ $hi = 30$ $mid = 15$ *jyada* $count = 8$ 

01:04:42 / 01:52:23



Ques : Koko eating bananas**[Leetcode 875]**
$$\begin{matrix} [30, 11, 23, 4, 20] & h=5 \\ \downarrow & \downarrow & \downarrow & \downarrow & \downarrow \\ 2 & 1 & 2 & 1 & 2 \end{matrix}$$

$K=?$

 $lo = 1 \ 16$ $hi = 30$ $mid = 18 \ 23$ $count = 8$ 

Ques : Koko eating bananas**[Leetcode 875]**

[30 , 11, 23, 4, 20]

↓ ↓ ↓ ↓ ↓

2 | | | |

$n = 5$

$K = ?$

$lo = 1 \ 16$

$hi = 30$

$mid = 18 \ 23$

$count = 8 \ 6$



Ques : Koko eating bananas**[Leetcode 875]**

$[30, 11, 23, 4, 20] \quad h=5$
↓ ↓ ↓ ↓ ↓
2 | | | | $K=?$

 $lo = 1 \ 16$ $hi = 30$ $mid = 18 \ 23$ $count = 8 \ 6$
 α 

Ques : Koko eating bananas**[Leetcode 875]**
$$\begin{bmatrix} 30, 11, 23, 4, 20 \end{bmatrix} \quad h=5$$
$$\begin{matrix} \downarrow & \downarrow & \downarrow & \downarrow & \downarrow \\ 2 & | & | & | & | \end{matrix} \quad k=?$$
 $lo = 1 \text{ to } 24$ $hi = 30$ $mid = 18 \text{ to } 23$ $count = 8 \text{ to } 6$ 

Ques : Koko eating bananas**[Leetcode 875]**

$$\left[\begin{matrix} 30, 11, 23, 4, 20 \end{matrix} \right] \quad h = 5$$
$$\begin{matrix} \downarrow & \downarrow & \downarrow & \downarrow & \downarrow \\ 2 & | & | & | & | \end{matrix} \quad k = ?$$

 $l_0 = 1 \text{ } 16 \text{ } 24$ $h_i = 30$ $mid = 18 \text{ } 23 \text{ } 27$ $count = 8 \text{ } 6$ 

Ques : Koko eating bananas

[Leetcode 875]

$$\begin{bmatrix} 30, 11, 23, 4, 20 \end{bmatrix}$$

↓ ↓ ↓ ↓ ↓
2 | | | |

$n = 5$
 $K = ?$

 $lo = 1 \times 6 24$ $hi = 30$ $mid = 18 \ 23 \ 27$ $count = 8 \ 6 \ 6$ 

Ques : Koko eating bananas**[Leetcode 875]**

$[30, 11, 23, 4, 20] \quad h=5$
↓ ↓ ↓ ↓ ↓
2 1 1 1 1 $K=?$

$lo = 1 \ 16 \ 24 \ 28$

$hi = 30$

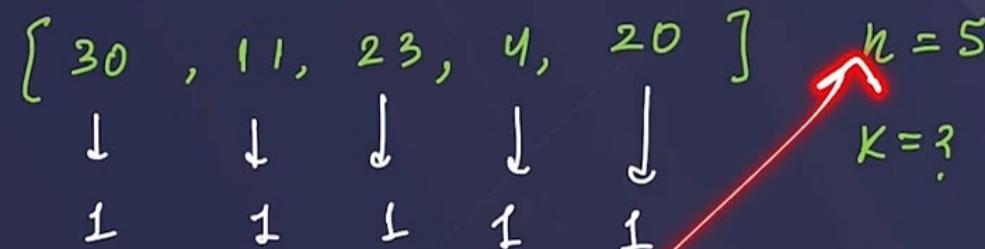
$mid = 18 \ 23 \ 24 \ 29$

$count = 8 \ 6 \ 8 \ 6$



Ques : Koko eating bananas

[Leetcode 875]

 $l_0 = 1 \ 16 \ 24 \ 28 \ 30$ $h_i = 30$ $mid = 18 \ 23 \ 24 \ 29 \ 30$ $count = 8 \ 6 \ 8 \ 6 \ 5$ 

01:07:13 / 01:52:23



Ques : Koko eating bananas**[Leetcode 875]**

$$\begin{array}{c} [30, 11, 23, 4, 20] \quad h=5 \\ \downarrow \quad \downarrow \quad \downarrow \quad \downarrow \quad \downarrow \\ 1 \quad 1 \quad 1 \quad 1 \quad 1 \end{array} \quad k=?$$

$$lo = 1 \cancel{16} \cancel{29} \cancel{28} \ 30$$

$$hi = 30$$

$$mid = 1\cancel{8} \cancel{23} \cancel{24} \cancel{29} \ 30$$

$$count = 8 \ 6 \cancel{8} \ \cancel{6} \ 5$$

$$ans = 30$$



Ques : Koko eating bananas

[Leetcode 875]

$$\begin{bmatrix} 30, 11, 23, 4, 20 \end{bmatrix} \quad h = 5$$
$$\begin{matrix} \downarrow & \downarrow & \downarrow & \downarrow & \downarrow \\ 1 & 1 & 1 & 1 & 1 \end{matrix}$$
$$K = ?$$
$$lo = 1 \ 16 \ 24 \ 28 \ 30$$
$$hi = 30 \ 29$$
$$mid = 18 \ 23 \ 24 \ 29 \ 30$$
$$count = 8 \ 6 \ \emptyset \ 6 \ 5$$
$$ans = 30$$


Ques : Koko eating bananas**[Leetcode 875]**

$\{ 30, 11, 23, 4, 20 \}$

↓ ↓ ↓ ↓ ↓
2 1 2 1 2

 $h = 6$ $lo = 1$ $hi = 30$ $mid = 15$ $count = 8$ $ans =$ 

01:08:32 / 01:52:23



Ques : Koko eating bananas**[Leetcode 875]**

$[30, 11, 23, 4, 20]$ $h=6$

↓ ↓ ↓ ↓ ↓
2 1 2 1 2

 $lo = 16$ $hi = 30$ $mid = 23$ $count = 8$ $ans =$ 

Ques : Koko eating bananas

[Leetcode 875]

 $lo = 16$ $hi = 30$ $mid = 18 \quad 23$ $count = 8 \quad 6$ $ans = 23$
$$\begin{matrix} [30, 11, 23, 4, 20] & h=6 \\ \downarrow & \downarrow & \downarrow & \downarrow & \downarrow \\ 2 & 1 & 1 & 1 & 1 \end{matrix}$$
 $count \leq hard$ 

01:09:11 / 01:52:23



Ques : Koko eating bananas**[Leetcode 875]**

$\{ 30, 11, 23, 4, 20 \}$ $h = 6$

↓ ↓ ↓ ↓ ↓
1 2 1 2 1 1

$lo = 1 \ 16 \ 20 \ 22 \ 23 \ 2 \quad 1 \quad 2 \quad 1 \quad 1$

$hi = 30 \ 22$

$count \leq hard$

$mid = 18 \ 23 \ 19 \ 21 \ 22$

$count = 8 \ 8 \ 7 \ 7$

$ans = 23$





Problem List

Premium

[Description](#) [Editorial](#) [Solutions](#) [Submissions](#)

C++ Auto

**Example 2:**

Input: piles = [30,11,23,4,20], h = 5
Output: 30

Example 3:

Input: piles = [30,11,23,4,20], h = 6
Output: 23

Constraints:

- $1 \leq \text{piles.length} \leq 10^4$
- $\text{piles.length} \leq h \leq 10^9$

Console ^



Run

Submit

```
1 class Solution {
2 public:
3     int minEatingSpeed(vector<int>& piles, int h) {
4         int n = piles.size();
5         int mx = -1;
6         for(int i=0;i<n;i++){
7             mx = max(mx,piles[i]);
8         }
9         int lo = 1;
10        int hi = mx;
11        int ans = -1;
12        while(lo<=hi){
13            int mid = lo + (hi-lo)/2;
14            if(check(mid,piles,h)==true){
15                ans = mid;
16                hi = mid - 1;
17            }
18        else lo = mid + 1;
19    }
20 }
```

Continue to work on your code from May 17, 2023 12:23:21 [Reopen](#)



Problem List

Premium



0

[Description](#) [Editorial](#) [Solutions](#) [Submissions](#)**Example 1:**

Input: piles = [3,6,7,11], h = 8
Output: 4

Example 2:

Input: piles = [30,11,23,4,20], h = 5
Output: 30

Example 3:

Input: piles = [30,11,23,4,20], h = 6
Output: 23

Console ^



Run

Submit

```
i C++ ▾   • Auto
1 class Solution {
2 public:
3     bool check(int speed, vector<int>& piles, int h){
4         int count = 0;
5         int n = piles.size();
6         for(int i=0;i<n;i++){
7             if(speed>=piles[i]) count++;
8             else if(piles[i]>speed) count += piles[i]/speed;
9             else count += piles[i]/speed + 1; // V.IMP
```

```
10        }
11        if(count>h) return false;
12        else return true;
13    }
14    int minEatingSpeed(vector<int>& piles, int h) {
15        int n = piles.size();
16        int mx = -1;
17        for(int i=0;i<n;i++){
18            mx = max(mx,piles[i]);
19        }
20        int lo = 1;
21        int hi = mx;
```

Continue to work on your code from May 17, 2023 12:23:21 [Reopen](#)



Problem List

Premium



0

[Description](#) [Editorial](#) [Solutions](#) [Submissions](#)**Example 1:**

Input: piles = [3,6,7,11], h = 8

Output: 4

Testcase

piles =
[30,11,23,4,20]

h =

5

Output

Console



Run

Submit

```
i C++ ▾   • Auto  
1 class Solution {  
2     public:  
3         bool check(int speed, vector<int>& piles, int h){  
4             int count = 0;  
5             int n = piles.size();  
6             for(int i=0;i<n;i++){  
7                 if(speed>=piles[i]) count++;  
8                 else if(piles[i]%speed==0) count += piles[i]/speed;  
9                 else count += piles[i]/speed + 1; // V.IMP  
10            }  
11            if(count>h) return false;  
12            else return true;  
13        }  
14        int minEatingSpeed(vector<int>& piles, int h) {  
15            int n = piles.size();  
16            int mx = -1;  
17            for(int i=0;i<n;i++){  
18                mx = max(mx,piles[i]);  
19            }  
20            int lo = 1;  
21            int hi = mx;
```

Continue to work on your code from May 17, 2023 12:23:21 [Reopen](#)



Problem List

Premium

0



Description Editorial Solutions Submissions

C++ Auto

**Example 1:**

Input: piles = [3,6,7,11], h = 8
Output: 4

Testcase Result

Accepted 4 ms

- Case 1
- Case 2
- Case 3

Input

piles =

Console



Run

Submit

```
1 class Solution {
2 public:
3     bool check(int speed, vector<int>& piles, int h){
4         int count = 0;
5         int n = piles.size();
6         for(int i=0;i<n;i++){
7             if(speed>=piles[i]) count++;
8             else if(piles[i]%speed==0) count += piles[i]/speed;
9             else count += piles[i]/speed + 1; // V.IMP
10        }
11        if(count>h) return false;
12        else return true;
13    }
14    int minEatingSpeed(vector<int>& piles, int h) {
15        int n = piles.size();
16        int mx = -1;
17        for(int i=0;i<n;i++){
18            mx = max(mx,piles[i]);
19        }
20        int lo = 1;
21        int hi = mx;
```

Continue to work on your code from May 17, 2023 12:23:21 [Reopen](#)



Problem List

Premium



0

[Description](#) [Editorial](#) [Solutions](#) [Submissions](#)**Example 1:**

Input: piles = [3,6,7,11], h = 8

Output: 4

Testcase

Last Executed Input

piles =

[805306368, 805306368, 805306368]

h =

1000000000

Console



Run

Submit

```
i C++ ▾   • Auto  
1 class Solution {  
2     public:  
3         bool check(int speed, vector<int>& piles, int h){  
4             int count = 0;  
5             int n = piles.size();  
6             for(int i=0;i<n;i++){  
7                 if(count>h) return false;  
8                 if(speed>=piles[i]) count++;  
9                 else if(piles[i]%speed==0) count += piles[i]/speed;  
10                else count += piles[i]/speed + 1; // V.IMP  
11            }  
12            if(count>h) return false;  
13            else return true;  
14        }  
15        int minEatingSpeed(vector<int>& piles, int h){  
16            int n = piles.size();  
17            int mx = -1;  
18            for(int i=0;i<n;i++){  
19                mx = max(mx,piles[i]);  
20            }  
21            int l = 1, r = mx + 1;
```

Continue to work on your code from May 17, 2023 12:23:21





Problem List

Premium



0

[Description](#) [Editorial](#) [Solutions](#) [Submissions](#)

Accepted

[Next question](#)[More challenges](#)

- [774. Minimize Max Distance to Gas Station](#)
- [2064. Minimized Maximum of Products Distributed to Any Store](#)
- [2498. Frog Jump II](#)

[All statuses](#)[All languages](#)

Raghav Garg

May 17, 2023 13:24

[Details](#)[+ Solution](#)

C++

Runtime 67 ms

Beats 26.38%

Memory 18.8 MB

Beats 99.82%

Click the distribution chart to view more details

Notes

Write your notes here

Related Tags

Select tags



Console



Run

Submit

01:21:11 / 01:52:23

class Solution {



0/5



LeetCode

Problem List Premium 0

Description Editorial Solutions Submissions

875. Koko Eating Bananas

Medium ✓ 7.6K 355

Companies

Koko loves to eat bananas. There are n piles of bananas, the i^{th} pile has $\text{piles}[i]$ bananas. The ...

Testcase Result

Runtime Error 121 / 125 testcases passed

```
Line 9: Char 46: runtime error: signed integer overflow: 1610612736 + 805306368 cannot be represented in type 'int' (solution.cpp)
SUMMARY: UndefinedBehaviorSanitizer: undefined-behavior prog_joined.cpp:18:46
```

Console ▾ Run Submit

C++ Auto

```
1 class Solution {
2 public:
3     bool check(int speed, vector<int>& piles, int h){
4         int count = 0;
5         int n = piles.size();
6         for(int i=0;i<n;i++){
7             // if(count>h) return false;
8             if(speed>=piles[i]) count++;
9             else if(piles[i]%speed==0) count += piles[i]/speed;
10            else count += piles[i]/speed + 1; // V.IMP
11        }
12        if(count>h) return false;
13        else return true;
14    }
15    int minEatingSpeed(vector<int>& piles, int h) {
16        int n = piles.size();
17        int mx = -1;
18        for(int i=0;i<n;i++){
19            mx = max(mx,piles[i]);
20        }
21        int l = 1, r = mx + 1;
```

Continue to work on your code from May 17, 2023 12:23:21



LeetCode

Problem List Premium 0

Description Editorial Solutions Submissions

875. Koko Eating Bananas

Medium ✓ 7.6K 355

Companies

Koko loves to eat bananas. There are n piles of bananas, the i^{th} pile has $\text{piles}[i]$ bananas. The...

Testcase Result

Runtime Error 121 / 125 testcases passed

Line 9: Char 46: runtime error: signed integer overflow: 1610612736 + 805306368 cannot be represented in type 'int' (solution.cpp)
SUMMARY: UndefinedBehaviorSanitizer: undefined-behavior prog_joined.cpp:18:46

Console Run Submit

i C++ Auto

```
3     bool check(int speed, vector<int>& piles, int h){  
4         long long count = 0;  
5         int n = piles.size();  
6         for(int i=0;i<n;i++){  
7             // if(count>h) return false;  
8             if(speed>=piles[i]) count++;  
9             else if(piles[i]%speed==0) count += (long long)(piles[i]  
/speed);  
10            else count += (long long)(piles[i]/speed + 1); // V.IMP  
11        }  
12        if(count>(long long)h) return false;  
13        else return true;  
14    }  
15    int minEatingSpeed(vector<int>& piles, int h) {  
16        int n = piles.size();  
17        int mx = -1;  
18        for(int i=0;i<n;i++){  
19            mx = max(mx,piles[i]);  
20        }  
21        int lo = 1;  
22        int hi = mx;
```

Continue to work on your code from May 17, 2023 12:23:21 Re X



LeetCode

Problem List

Premium

0

Accepted

Raghav Garg

May 17, 2023 13:25

C++

Runtime 78 ms Beats 10.90% Memory 18.9 MB Beats 90.22%

Click the distribution chart to view more details

Notes

Write your notes here

All statuses

All languages

Console ^ Run Submit

Related Tags

Select tags

class Solution {

0/5



Ques : Minimum time to complete trips [Leetcode 2187]





Ques : Minimum time to complete trips [Leetcode 2187]



Ques : Minimum time to complete trips [Leetcode 2187]

B1 B2 B3

[1 2 3]

totalTrips = 5

After 1 hour

[1 0 0]

minTime =

After 2 hours

[2 1 0]

After 3 hours [



Ques : Minimum time to complete trips [Leetcode 2187]

B1 B2 B3

[1 2 3]

total Trips = 5

After 1 hour

[1 0 0]

1 minTime =

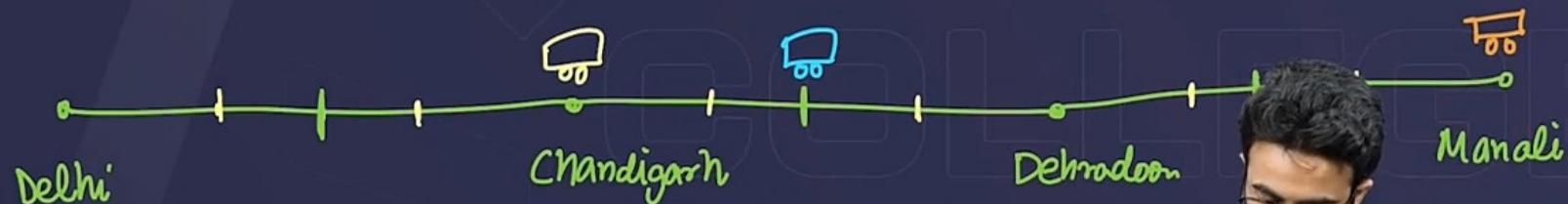
After 2 hours

[2 1 0]

2.

After 3 hours

[3 1 1]



Ques : Minimum time to complete trips [Leetcode 2187]

B1	B2	B3
1	2	3

atleast
total Trips = 5

After 1 hour [1 0 0] 1 minTime =

After 2 hours [2 1 0] 3

After 3 hours [3 1 1] 5

After 4 hours [4 2 1] 7

Lucknow
100



Delhi



01:34:14 / 01:52:23 Chandigarh

Dehradoon

SKILLS



Ques : Minimum time to complete trips [Leetcode 2187]

	B1	B2	B3	atleast total Trips = 5	
	[1]	2	3]		
After 1 hour	[1 0 0]	10 / 3.	minTime =	[10, 6, 3]	now
After 2 hours	[2 1 0]				
After 3 hours	[3 1 1]	5			
After 4 hours	[4 2 1]	7			



Delhi



01:36:46 / 01:52:23

Chandigarh

Dehradoon



Ques : Minimum time to complete trips [Leetcode 2187]

atleast total Trips = 5

After 1 hour $[1, 0, 0]$ minTime = $10 / 3$

After 2 hours $[2, 1, 0]$

After 3 hours $[3, 1, 1]$ 5

After 4 hours $[4, 2, 1]$ 7

$10 + 3 = 13$

$13 + 3 = 16$

$16 + 3 = 19$

$19 + 3 = 22$

$22 + 3 = 25$

$25 + 3 = 28$

$28 + 3 = 31$

$31 + 3 = 34$

$34 + 3 = 37$

$37 + 3 = 40$

$40 + 3 = 43$

$43 + 3 = 46$

$46 + 3 = 49$

$49 + 3 = 52$

$52 + 3 = 55$

$55 + 3 = 58$

$58 + 3 = 61$

$61 + 3 = 64$

$64 + 3 = 67$

$67 + 3 = 70$

$70 + 3 = 73$

$73 + 3 = 76$

$76 + 3 = 79$

$79 + 3 = 82$

$82 + 3 = 85$

$85 + 3 = 88$

$88 + 3 = 91$

$91 + 3 = 94$

$94 + 3 = 97$

$97 + 3 = 100$



Delhi



01:36:50 / 01:52:23

Chandigarh

Dehradoon



Ques : Minimum time to complete trips [Leetcode 2187]

atleast total Trips = 5

After 1 hour $[1, 0, 0]$ minTime = $10/3$

After 2 hours $[2, 1, 0]$

After 3 hours $[3, 1, 1]$ 5

After 4 hours $[4, 2, 1]$ 7

$10 + 3 = 13$

$13 + 3 = 16$

$16 + 3 = 19$

$19 + 3 = 22$

$22 + 3 = 25$

$25 + 3 = 28$

$28 + 3 = 31$

$31 + 3 = 34$

$34 + 3 = 37$

$37 + 3 = 40$

$40 + 3 = 43$

$43 + 3 = 46$

$46 + 3 = 49$

$49 + 3 = 52$

$52 + 3 = 55$

$55 + 3 = 58$

$58 + 3 = 61$

$61 + 3 = 64$

$64 + 3 = 67$

$67 + 3 = 70$

$70 + 3 = 73$

$73 + 3 = 76$

$76 + 3 = 79$

$79 + 3 = 82$

$82 + 3 = 85$

$85 + 3 = 88$

$88 + 3 = 91$

$91 + 3 = 94$

$94 + 3 = 97$

$97 + 3 = 100$



01:36

Drag from top and touch the back button to exit full screen.



Ques : Minimum time to complete trips [Leetcode 2187]

	B1	B2	B3	
After 1 hour	[1]	0	0]	$10 / 3$
After 2 hours	[2]	1	0]	$10 / 3$
After 3 hours	[3]	1	1]	5
After 4 hours	[4]	2	1]	7

atleast total Trips = 5

minTime = $10 + 3$

18 miles



Delhi



01:36:54 / 01:52:23 Chandigarh

Dehradoon





Problem List

Premium

[Description](#) [Editorial](#) [Solutions](#) [Submissions](#)

2187. Minimum Time to Complete Trips

[Hint](#)

Medium



2.5K



153



Companies

You are given an array `time` where `time[i]` denotes the time taken by the i^{th} bus to complete **one trip**.

Each bus can make multiple trips **successively**; that is, the next trip can start **immediately after** completing the current trip. Also, each bus operates **independently**; that is, the trips of one bus do not influence the trips of any other bus.

You are also given an integer `totalTrips`, which denotes the number of trips all buses should make in

Console ^



Run

Submit

```
i C++ ▾   • Auto  
1 class Solution {  
2 public:  
3     long long minimumTime(vector<int>& time, int totalTrips) {  
4         int n = time.size();  
5         long long lo = 1;  
6         long long hi = ?  
7         long long ans = -1;  
8         while(lo<=hi){  
9             long long mid = lo + (hi-lo)/2;  
10            if(check(mid, time, totalTrips)==true){  
11                ans = mid;  
12                hi = mid - 1;  
13            }  
14            else lo = mid + 1;  
15        }  
16    }  
17 }
```





Problem List

Premium

[Description](#) [Editorial](#) [Solutions](#) [Submissions](#)

2187. Minimum Time to Complete Trips

[Hint](#)

Medium



2.5K



153



Companies

You are given an array `time` where `time[i]` denotes the time taken by the i^{th} bus to complete **one trip**.

Each bus can make multiple trips **successively**; that is, the next trip can start **immediately after** completing the current trip. Also, each bus operates **independently**; that is, the trips of one bus do not influence the trips of any other bus.

You are also given an integer `totalTrips`, which denotes the number of trips all buses should make in

```
i C++ ▾   • Auto  
1 class Solution {  
2 public:  
3     long long minimumTime(vector<int>& time, int totalTrips) {  
4         int n = time.size();  
5         long long lo = 1;  
6         long long hi = ?;  
7         long long ans = -1;  
8         while(lo<=hi){  
9             long long mid = lo + (hi-lo)/2;  
10            if(check(mid, time, totalTrips)==true){  
11                ans = mid;  
12                hi = mid - 1;  
13            }  
14            else lo = mid + 1;  
15        }  
16        return ans;  
17    }  
18};
```



01:38:28 / 01:52:23





Problem List

Premium

[Description](#) [Editorial](#) [Solutions](#) [Submissions](#)

You are also given an integer `totalTrips`, which denotes the number of trips all buses should make in **total**. Return the **minimum time** required for all buses to complete **at least** `totalTrips` trips.

Example 1:

Input: `time = [1,2,3]`, `totalTrips = 5`
Output: 3

Explanation:

- At time $t = 1$, the number of trips completed by each bus are $[1, 0, 0]$.
The total number of trips completed is $1 + 0 + 0 = 1$.
- At time $t = 2$, the number of trips completed by each bus are $[2, 1, 0]$.
The total number of trips completed

Console ^



Run

Submit

```
i C++ ▾   • Auto  
1 class Solution {  
2 public:  
3     long long minimumTime(vector<int>& time, int totalTrips) {  
4         // 3 3 3  
5         // after 15 hours [5 5 5] = 15  
6         int n = time.size();  
7         for(int i=0;i<n;i++){  
8             mx = max(mx,time[i]);  
9         }  
10        long long lo = 1;  
11        long long hi = (long long)mx*(long long)totalTrips/(long  
12        long)n;  
13        long long ans = -1;  
14        while(lo<=hi){  
15            long long mid = lo + (hi-lo)/2;  
16            if(check(mid, time, totalTrips)){  
17                ans = mid;  
18                hi = mid - 1;  
19            }  
20            else lo = mid + 1;  
21        }  
        return ans;  
    }
```





Problem List

Premium



```
i C++ ▾   • Auto
1 class Solution {
2 public:
3     long long minimumTime(vector<int>& time, int totalTrips) {
4         // 3 3 3
5         // after 15 hours [5 5 5] = 15
6         int n = time.size();
7         for(int i=0;i<n;i++){
8             mx = max(mx,time[i]);
9         }
10        long long lo = 1;
11        long long hi = (long long)mx*(long long)totalTrips/(long long)n;
12        long long ans = -1;
13        while(lo<=hi){
14            long long mid = lo + (hi-lo)/2;
15            if(check(mid, time, totalTrips)==true){
16                ans = mid;
17                hi = mid - 1;
18            }
19            else lo = mid + 1;
20        }
21        return ans;
22    }
23};
```





Problem List

Premium



```
i C++ ▾   • Auto
1 class Solution {
2 public:
3     bool check(long long mid, vector<int>& time, int totalTrips){
4         long long trips = 0;
5         int n = time.size();
6         // 1 2 3   midhours = 2
7         for(int i=0;i<n;i++){
8             trips += mid/(long long)time[i];
9         }
10        if(trips<(long long)totalTrips) return false;
11        else return true;
12    }
13    long long minimumTime(vector<int>& time, int totalTrips) {
14        // 3 3 3
15        // after 15 hours [5 5 5] = 15
16        int n = time.size();
17        for(int i=0;i<n;i++){
18            mx = max(mx,time[i]);
19        }
20        long long lo = 1;
21        long long hi = (long long)mx*(long long)totalTrips/(long long)n;
22        long long ans = -1;
23        while(lo<=hi){
```





Problem List

Premium

0

[Description](#) [Editorial](#) [Solutions](#) [Submissions](#)

You are also given an integer `totalTrips`, which denotes the number of trips all buses should make in total. Return the **minimum time** required for all buses to complete **at least** `totalTrips` trips.

Example 1:

...

[Testcase](#) [Result](#)

Judging...



Speed Up



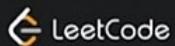
Run

Submit

C++ ▾ Auto

```
11     else return true;
12 }
13 long long minimumTime(vector<int>& time, int totalTrips) {
14     // 3 3 3
15     // after 15 hours [5 5 5] = 15
16     int n = time.size();
17     for(int i=0;i<n;i++){
18         mx = max(mx,time[i]);
19     }
20     long long lo = 1;
21     long long hi = (long long)mx*(long long)totalTrips/(long
22     long)n;
23     long long ans = -1;
24     while(lo<=hi){
25         long long mid = lo + (hi-lo)/2;
26         if(check(mid, time, totalTrips))
27             ans = mid;
28         else lo = mid + 1;
29     }
30 }
31
32 }
```





Problem List

Premium

[Description](#) [Editorial](#) [Solutions](#) [Submissions](#)

You are also given an integer `totalTrips`, which denotes the number of trips all buses should make in total. Return the **minimum time** required for all buses to complete **at least** `totalTrips` trips.

Example 1:

[Testcase](#) [Result](#)

Compile Error

```
Line 18: Char 13: error: use of undeclared identifier 'mx'  
        mx = max(mx,time[i]);  
        ^
```

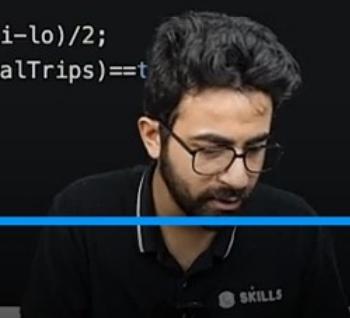
C++ ▾ Auto

```
11     else return true;  
12 }  
13 long long minimumTime(vector<int>& time, int totalTrips) {  
14     // 3 3 3  
15     // after 15 hours [5 5 5] = 15  
16     int n = time.size();  
17     int mx = -1;  
18     for(int i=0;i<n;i++){  
19         mx = max(mx,time[i]);  
20     }  
21     long long lo = 1;  
22     long long hi = (long long)mx*(long long)totalTrips/(long  
long)n;  
23     long long ans = -1;  
24     while(lo<=hi){  
25         long long mid = lo + (hi-lo)/2;  
26         if(check(mid, time, totalTrips)==t  
27             ans = mid;  
28             hi = mid - 1;  
29         }  
30         else lo = mid + 1;  
31     }  
32     return ans;
```



01:47:42 / 01:52:23

Submit





Problem List

Premium

[Description](#) [Editorial](#) [Solutions](#) [Submissions](#)

You are also given an integer `totalTrips`, which denotes the number of trips all buses should make in total. Return the **minimum time** required for all buses to complete **at least** `totalTrips` trips.

Example 1:

Testcase

Result

Accepted 4 ms

Case 1 Case 2

Input

time =

C++ Auto

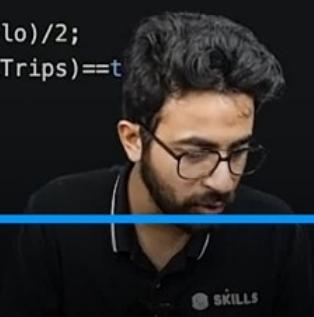
```
11     else return true;
12 }
13 long long minimumTime(vector<int>& time, int totalTrips) {
14     // 3 3 3
15     // after 15 hours [5 5 5] = 15
16     int n = time.size();
17     int mx = -1;
18     for(int i=0;i<n;i++){
19         mx = max(mx,time[i]);
20     }
21     long long lo = 1;
22     long long hi = (long long)mx*(long long)totalTrips/(long
long)n;
23     long long ans = -1;
24     while(lo<=hi){
25         long long mid = lo + (hi-lo)/2;
26         if(check(mid, time, totalTrips)==t
27             ans = mid;
28             hi = mid - 1;
29         } else lo = mid + 1;
30     }
31 }
32 return ans;
```



01:47:49 / 01:52:23

Run

Submit





Problem List

Premium

[Description](#) [Editorial](#) [Solutions](#) [Submissions](#)

You are also given an integer `totalTrips`, which denotes the number of trips all buses should make in total. Return the **minimum time** required for all buses to complete **at least** `totalTrips` trips.

Example 1:

Testcase

...



Result

Wrong Answer

Input

time =

[9,7,10,9,10,9,10]

Use Testcase

C++ Auto

```
11     else return true;
12 }
13 long long minimumTime(vector<int>& time, int totalTrips) {
14     // 3 3 3
15     // after 15 hours [5 5 5] = 15
16     int n = time.size();
17     int mx = -1;
18     for(int i=0;i<n;i++){
19         mx = max(mx,time[i]);
20     }
21     long long lo = 1;
22     long long hi = (long long)mx*(long long)totalTrips/(long
long)n;
23     long long ans = -1;
24     while(lo<=hi){
25         long long mid = lo + (hi-lo)/2;
26         if(check(mid, time, totalTrips))
27             ans = mid;
28         else hi = mid - 1;
29     }
30     else lo = mid + 1;
```



Console



01:47:57 / 01:52:23





Problem List

Premium

[Description](#) [Editorial](#) [Solutions](#) [Submissions](#)

You are also given an integer `totalTrips`, which denotes the number of trips all buses should make in total. Return the **minimum time** required for all buses to complete **at least** `totalTrips` trips.

Example 1:[Testcase](#) [Result](#)

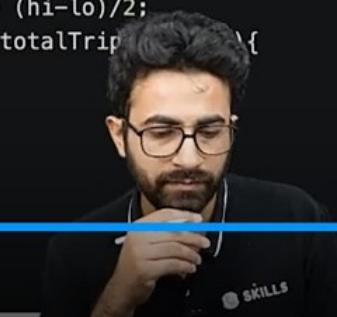
C++ ▾ Auto

```
11     else return true;
12 }
13 long long minimumTime(vector<int>& time, int totalTrips) {
14     // 3 3 3
15     // after 15 hours [5 5 5] = 15
16     int n = time.size();
17     int mx = -1;
18     for(int i=0;i<n;i++){
19         mx = max(mx,time[i]);
20     }
21     long long lo = 1;
22     long long hi = (long long)mx*(long long)totalTrips;
23     long long ans = -1;
24     while(lo<=hi){
25         long long mid = lo + (hi-lo)/2;
26         if(check(mid, time, totalTrips)) {
27             ans = mid;
28             hi = mid - 1;
29         } else lo = mid + 1;
30     }
31 }
32 return ans;
33 }
```



Pending... Run Submit

01:48:27 / 01:52:28



LeetCode

Problem List Premium

Description Editorial Solutions Submissions

Raghav Garg May 17, 2023 14:38 Details + Solution

C++

Runtime 479 ms Beats 28.17% Memory 94.5 MB Beats 44.70%

Click the distribution chart to view more details

All statuses All languages

Accepted C++ a few seconds ago

Notes Write your notes here

Related Tags Select tags 0/5

Console Run Submit

class Solution {

Ques : Minimum time to complete trips

[Leetcode 2187]

B1 B2 B3 B4 B5 B6 B7
[9, 7, 10, 9, 10, 9, 10]

```
long long minimumTime(vector<int>& time, int totalTrips) {
    // 3 3 3
    // after 15 hours [5 5 5] = 15
    int n = time.size();
    int mx = -1;
    for(int i=0;i<n;i++){
        mx = max(mx,time[i]); mx = 10
    }
    long long lo = 1;
    long long hi = (long long)mx*(long long)totalTrips/(long long)n;
    long long ans = -1;
    while(lo<=hi){
        long long mid = lo + (hi-lo)/2;
        if(check(mid, time, totalTrips)==true){
            ans = mid;
            hi = mid - 1;
        }
        else lo = mid + 1;
    }
    return ans;
}
```

$$\begin{aligned} \text{totalTrips} &= 1 \\ \text{minTime} &=? \end{aligned}$$

$$\begin{aligned} lo &= 1 \\ hi &= mx * tt / n = 10 * 1 / 7 \\ &= 1 \end{aligned}$$



Ques : Minimum time to complete trips

[Leetcode 2187]

 $B_1 \ B_2 \ B_3 \ B_4 \ B_5 \ B_6 \ B_7$ $[9, 7, 10, 9, 10, 9, 10]$

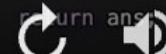
```
long long minimumTime(vector<int>& time, int totalTrips) {  
    // 3 3 3  
    // after 15 hours [5 5 5] → 15  
    int n = time.size();  
    int mx = -1;  
    for(int i=0;i<n;i++){  
        mx = max(mx,time[i]);  
    }  
    long long lo = 1;  
    long long hi = (long long)mx*(long long)totalTrips/(long long)n;  
    long long ans = -1;  
    while(lo<=hi){  
        long long mid = lo + (hi-lo)/2;  
        if(check(mid, time, totalTrips)==true){  
            ans = mid;  
            hi = mid - 1;  
        }  
        else lo = mid + 1;  
    }  
    return ans;  
}
```

$$\text{totalTrips} = 1$$

$$\minTime = ?$$

$$\frac{mx * tt}{n} \rightarrow 1$$

$$= 1$$



01:51:05 / 01:52:23



Ques : Minimum time to complete trips

[Leetcode 2187]

B1 B2 B3 B4 B5 B6 B7

[9, 7, 10, 9, 10, 9, 10]

```
long long minimumTime(vector<int>& time, int totalTrips) {  
    // 3 3 3  
    // after 15 hours [5 5 5] 15  
    int n = time.size();  
    int mx = -1;  
    for(int i=0;i<n;i++){  
        mx = max(mx,time[i]);  
    }  
    long long lo = 1;  
    long long hi = (long long)mx*(long long)totalTrips/(long long)n;  
    long long ans = -1;  
    while(lo<=hi){  
        long long mid = lo + (hi-lo)/2;  
        if(check(mid, time, totalTrips)==true){  
            ans = mid;  
            hi = mid - 1;  
        }  
        else lo = mid + 1;  
    }  
    return ans;  
}
```

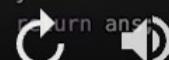
totalTrips = 1

minTime = ?

$$\text{minTime} = \frac{\text{mx} * \text{totalTrips}}{\text{n}} = \frac{10 * 1}{7}$$

$$= 1$$

Drag from top and touch the back button to exit full screen.



01:51:06 / 01:52:23



Ques : Minimum time to complete trips

[Leetcode 2187]

B1 B2 B3 B4 B5 B6 B7
 $[9, 7, 10, 9, 10, 9, 10]$

```
long long minimumTime(vector<int>& time, int totalTrips) {
    // 3 3 3
    // after 15 hours [5 5 5] = 15
    int n = time.size();
    int mx = -1;
    for(int i=0;i<n;i++){
        mx = max(mx,time[i]); mx = 10
    }
    long long lo = 1;
    long long hi = (long long)mx*(long long)totalTrips/(long long); 10 * 1 / 7
    long long ans = -1;
    while(lo<=hi){
        long long mid = lo + (hi-lo)/2;
        if(check(mid, time, totalTrips)==true){
            ans = mid;
            hi = mid - 1;
        }
        else lo = mid + 1;
    }
    return ans;
}
```

$$\begin{aligned} \text{totalTrips} &= 1 \\ \text{minTime} &=? \end{aligned}$$

$$lo = 1$$

$$\begin{aligned} hi &= mx * tt / n = 10 * 1 / 7 \\ &= 1 \end{aligned}$$



Ques : Capacity to ship packages within D days
[Leetcode 10]





Ques : Capacity to ship packages within D days
[Leetcode 1011]



Ques : Capacity to ship packages within D days [Leetcode 1011]

weights = { 1, 2, 3, 4, 5, 6, 7, 8, 9, 10 } day8 = 5

Capacity ⌈ ⌉ ship → minimum

1 Locatio → loc
ship



Ques : Capacity to ship packages within D days [Leetcode 1011]

weights = {1, 2, 3, 4, 6, 7, 8, 9, 10} day8 = 5

Capacity of ship → minimum
 $2x \rightarrow 20$

Day-1 → 1, 2, 3, 4, 5

Ques : Capacity to ship packages within D days [Leetcode 1011]

weights = { 1, 2, 3, 4, 5, 6, 7, 8, 9, 10 } day8 = 5

Capacity of ship → minimum

$$2x \rightarrow 20$$

Day-1 → 1, 2, 3, 4, 5 : 15 ✓

Day-2 → 6, 7 : 13 ✓

Day-3 → 8, 9 : 17 ✓

Day-4 → 10 : 10 ✓



Ques : C

Cut

Copy

Delete

Resize

Color

Take Screenshot

Add Element

Convert

7 days**[Leetcode 1011]**

weights = { 1, 2, 3, 4, 5, 6, 7, 8, 9, 10 } day8 = 5

Capacity of ship → minimum

$$2x \rightarrow 20$$

Day-1 → 1, 2, 3, 4, 5 : 15 ✓

Day-2 → 6, 7 : 13 ✓

Day-3 → 8, 9 : 17 ✓

Day-4 → 10 : 10 ✓



Ques : Capacity to ship packages within D days [Leetcode 1011]

weights = { 1, 2, 3, 4, 5, 6, 7, 8, 9, 10 } day8 = 5

Capacity of ship \rightarrow minimum

2x \rightarrow 10

Day 1 : 1, 2, 3, 4 : 10 ✓

Day 2 : 4, 5 : 9 ✓

Day 3 : 6

Day 2 : 5
Day 3 : 6



Ques : Capacity to ship packages within D days

[Leetcode 1011]

weights = { 1, 2, 3, 4, 5, 6, 7, 8, 9, 10 }

day8 = 5.

↓
Reh gaye

Capacity of ship → minimum

2x → 10

Day 1 : 1, 2, 3, 4 : 10 ✓

Day 2 : 4, 5 : 9 ✓

Day 3 : 6 : 6 ✓

Day - 4 : 7 ✓

Day - 5 : 8 ✓



Ques : Capacity to ship packages within D days

[Leetcode 1011]

weights = { 1, 2, 3, 4, 5, 6, 7, 8, 9, 10 } ? day 8 = 8 + ~~+~~

Capacity of ship \rightarrow minimum

↓
Reh gaye

$2x \rightarrow 10$

Day 1 : 1, 2, 3, 4 : 10 ✓

Day 2 : 4, 5 : 9 ✓

Day 3 : 6 : 6 ✓

Day - 4 : 7 ✓

Day - 5 : 8 ✓ Dua



Ques : Capacity to ship packages within D days

[Leetcode 1011]

weights = { 1, 2, 3, 4, 5, 6, 7, 8, 9, 10 } day8 = 5

Capacity of ship → minimum

2x → 10

Day 1 : 1, 2, 3, 4 : 10 ✓

Day 2 : 4, 5 : 9 ✓

Day 3 : 6 : 6 ✓

Day - 4 : 7 ✓

Day - 5 : 8 ✓

Capacity : 15

Day 1 → 1, 2, 3, 4, 5

Day 2 → 6, 7

Day 3 → 8

Day 4 → 9

Day 5 → 10



Ques : Capacity to ship packages within D days

[Leetcode 1011]

weights = { 1, 2, 3, 4, 5, 6, 7, 8, 9, 10 } day8 = 5

Capacity of ship \rightarrow minimum

Capacity : 15

Day 1 \rightarrow 1, 2, 3, 4, 5

Day 2 \rightarrow 6, 7

Day 3 \rightarrow 8

Day 4 \rightarrow 9

Day 5 \rightarrow 10



Ques : Capacity to ship packages within D days [Leetcode 1011]

weights = { 1, 2, 3, 4, 5, 6, 7, 8, 9, 10 } days = 5

Capacity of ship → minimum

Capacity → 14

Day 1 : 1, 2, 3, 4

Day 2 : 5, 6

Day 3 : 7

Day 4 : 8

Day 5 : 9



Ques : Capacity to ship packages within D days

[Leetcode 1011]

weights = { 3, 2, 2, 4, 1, 4 }

days = 3

min^m Capacity :

Capacity = 16

Day 1 → 3, 2, 2, 4, 1, 4



Ques : Capacity to ship packages within D days

[Leetcode 1011]

weights = { 3, 2, 2, 4, 1, 4 }

days = 3

min^m Capacity :

Capacity = 16

Day 1 → 3, 2, 2, 4, 1, 4

B-S → m 4 to 16
 lo hi

lo = max element of array

mi = sum of array

Kaam

no paa raha hai



Ques : Capacity to ship packages within D days

[Leetcode 1011]

weights = {3, 2, 2, 4, 1, 4, 3}

days = 3

sum = 16, max = 4

O(n) time complexity

