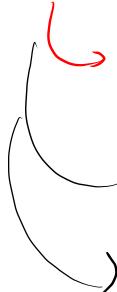


ArrayList

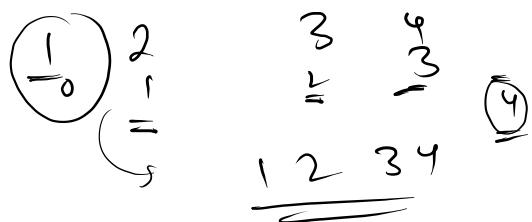


Dynamic representation of Array

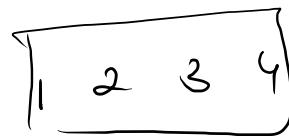
we can store same kind of data.

don't define size.

- initialize
- add()
- remove()
- get()
- size()



arr.get(i)



The banana challenge

Problem

Submissions

Leaderboard

Discussions

$$\underline{\text{banana/hour.}} = k$$

Koko is fond of consuming bananas and is faced with n piles of bananas, where the i th pile has $\text{piles}[i]$ bananas. Meanwhile, the guards have temporarily left and are expected to return in h hours.

Koko has the freedom to determine her banana-eating speed per hour, which she can set to k . Every hour, she selects a pile of bananas and consumes k bananas from that pile. However, if the selected pile has less than k bananas, she finishes all the bananas in that pile and won't eat any more bananas in that hour.

$$n=8 \rightarrow 8 \text{ hours}$$

Koko prefers to eat slowly but is still determined to finish consuming all the bananas before the guards come back.

Return the minimum integer k such that she can eat all the bananas within h hours.

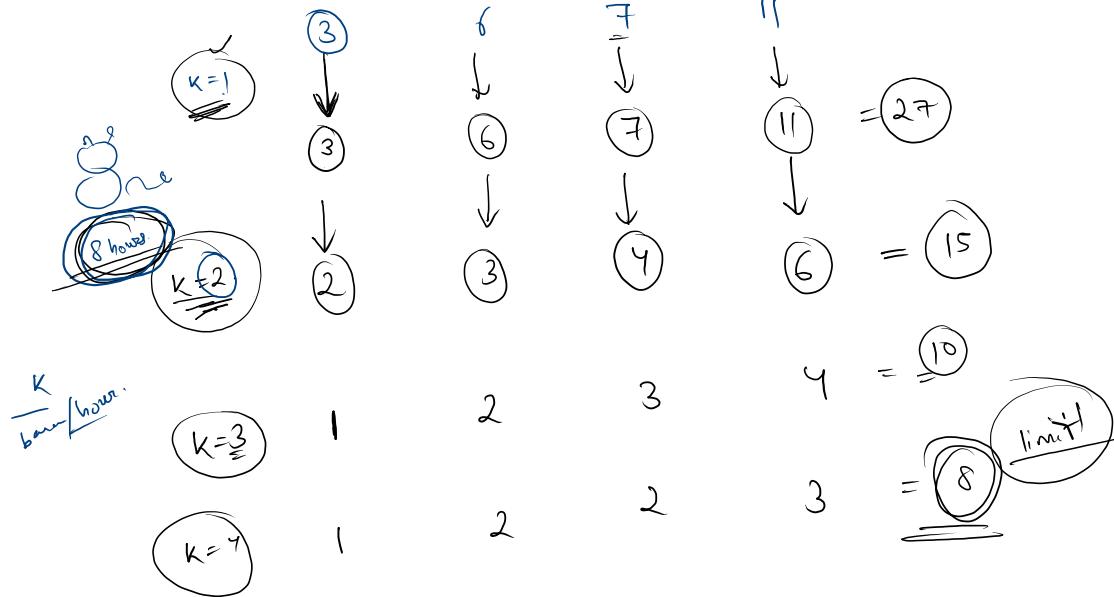
Sample Input 0

4
3 6 7 11
8

$$\underline{\text{banana/hour.}}$$

Sample Output 0

0 4 $\leftarrow k$



$k = \text{banana/hour}$

1 banana per hour

$$\underline{\underline{k=1}}$$

$$\underline{\underline{k=2}}$$

$$\underline{\underline{k=3}}$$

$$\underline{\underline{k=4}}$$

$$\underline{\underline{k=5}}$$

$$\underline{\underline{k=6}}$$

3	6	7	11	27
3	6	7	11	$\rightarrow \underline{\underline{27}}$
2	3	4	5	$\rightarrow \underline{\underline{15}}$
1	2	3	4	$\rightarrow \underline{\underline{10}}$

$$27 \leq 8$$

$$\checkmark \quad 1 \quad 2 \quad 2 \quad 3 \quad \rightarrow \underline{\underline{8}}$$

1	2	2	3
1	1	2	2

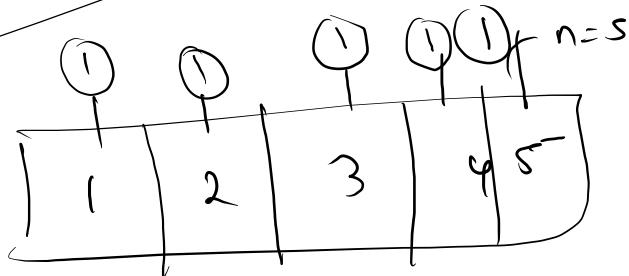
n

$\leq h \cup$

pile.length

at least

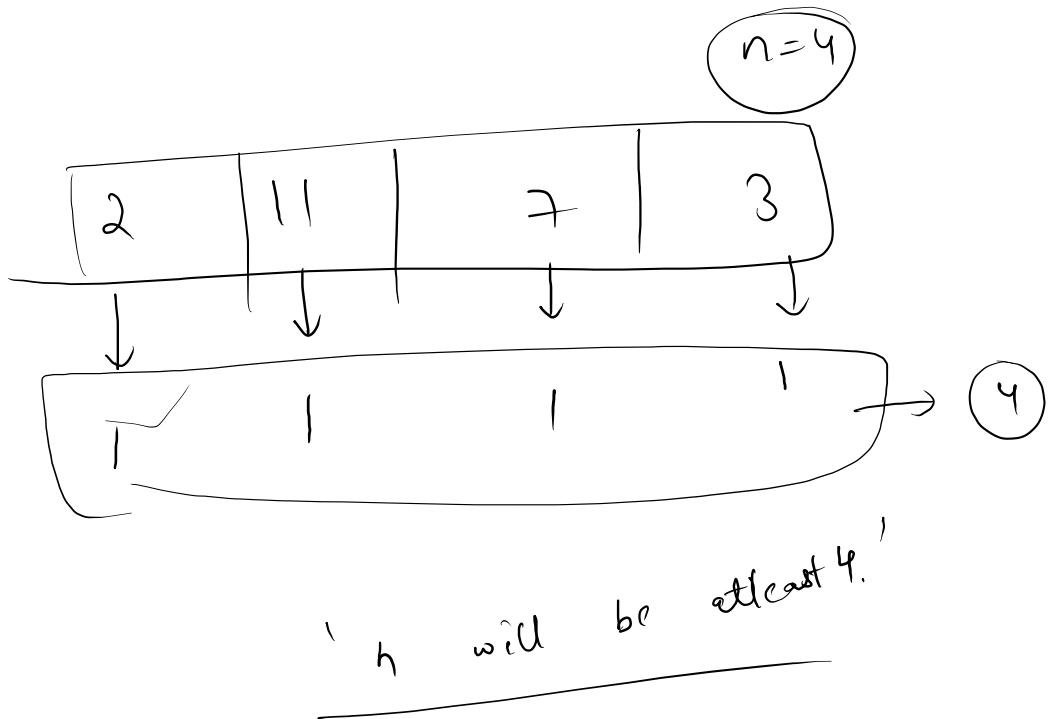
$h = 5.$



$\hookrightarrow h=5$

Maximum possible answer ('k') =
=

$k = 5.$



low = 1

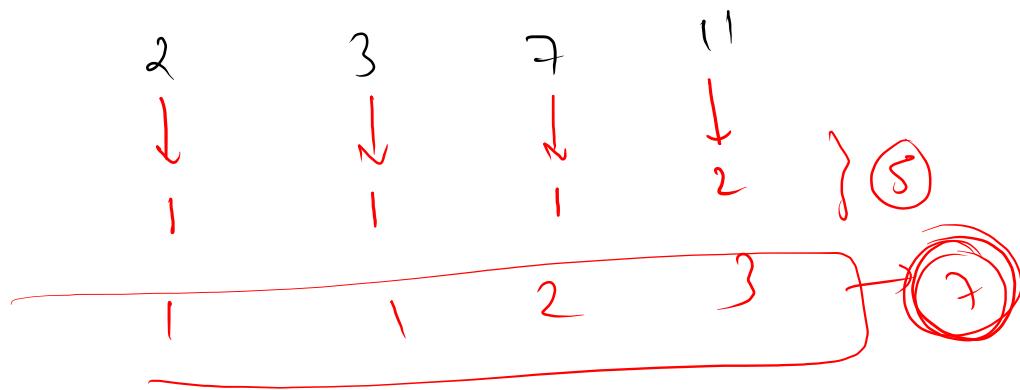
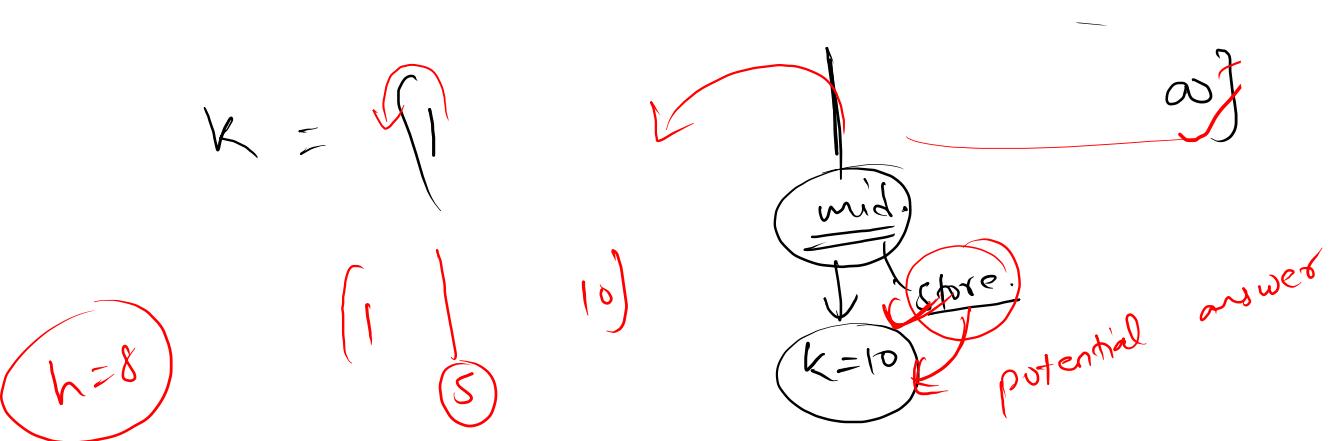
high = max ele of array.

Search space range of K

max. ele of array,

sorted range.

BS.



```
public static void main(String[] args) {  
    Scanner scn = new Scanner(System.in);  
    int n = scn.nextInt();  
    int [] A= new int[n];  
    for(int i = 0; i < n; i++){  
        A[i] = scn.nextInt();  
    }  
    int h = scn.nextInt(); //given hours
```

is also const.

The painter

Problem

Submissions

Leaderboard

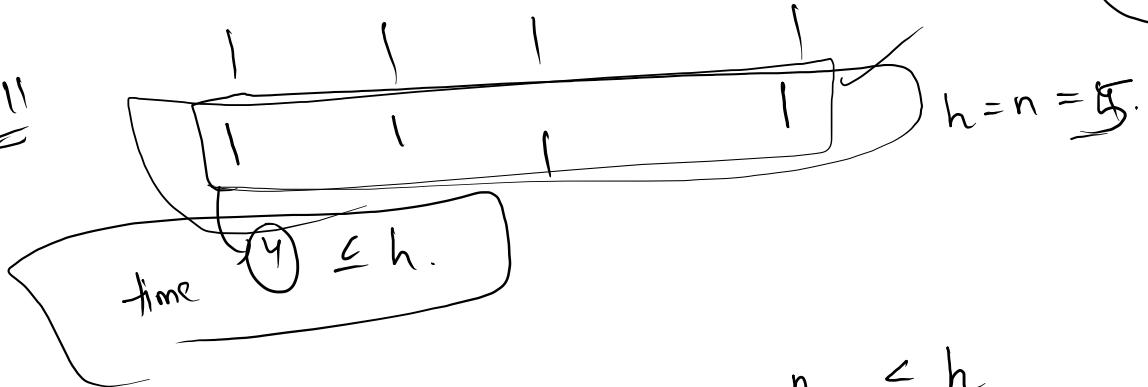
Discussions

We have to paint n boards of length {A1, A2...An}. There are k painters available and each takes 1 unit of time to paint 1 unit of the board. The problem is to find the minimum time to get this job was done under the constraints that any painter will only paint continuous sections of boards, say board {2, 3, 4} or only board {1} or nothing but not board {2, 4, 5}.

```
        ...  
    }  
    else{  
        low = mid + 1;  
    }  
}  
System.out.println(ans);  
  
public static boolean isPossible(int [] A, int h, int k){  
    int time = 0;  
  
    for(int ele : A){  
        time += Math.ceil(ele * 1.0 / k);  
    }  
  
    return time <= h;  
}
```

$$k=11$$

2 3 7 11



$$n \leq h$$

The painter

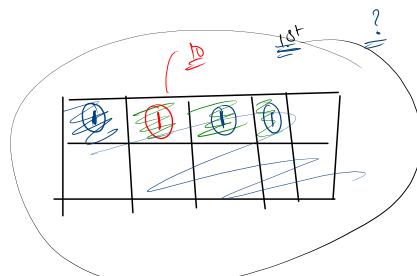
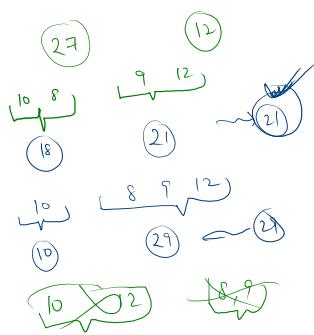
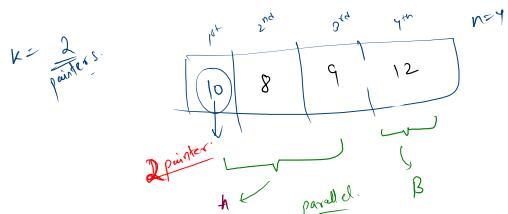
Problem

Submissions

Leaderboard

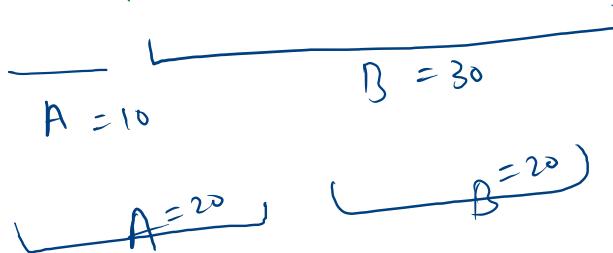
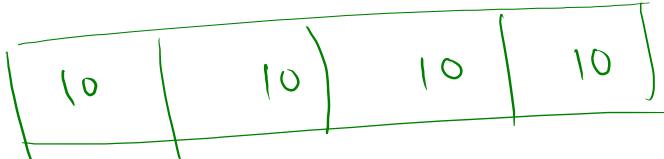
Discussions

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 the board. The problem is to find the minimum time to get this job done under the constraints that any painter will only paint continuous sections of boards, say board $(2, 3, 4)$ or only board (1) or nothing but not board $(2, 4, 5)$.

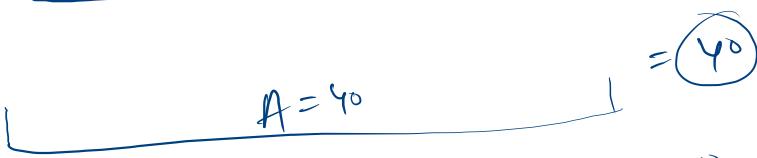
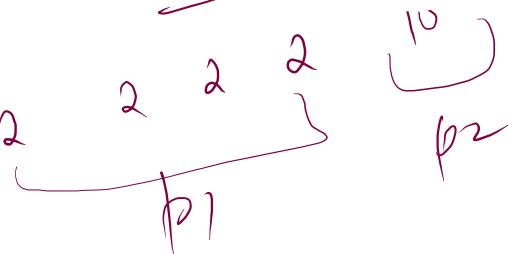


$k=2$

$n=4$



$k=3$



k=3

$$A \quad B \quad C = 20$$

Diagram showing four segments labeled A, B, C, and D. Segments A, B, and C each have a length of 10, indicated by blue brackets above them. Segment D is represented by a green bracket below it. The total length of segments A, B, and C is 30, indicated by a green circle with a checkmark.

$$A \quad B \quad C = 20$$

Diagram showing three segments labeled A, B, and C. Segments A and B are grouped together by a green bracket above them, indicating they have a combined length of 20. Segment C is represented by a green bracket below it. The total length of segments A, B, and C is 30, indicated by a green circle with a checkmark.

k=4

$$A \quad B \quad C \quad D = 40$$

Diagram showing four segments labeled A, B, C, and D. Each segment is underlined, indicating they all have equal lengths. The total length of all four segments is 40, indicated by a green circle with a checkmark.

K=1

$$\overbrace{10} + \overbrace{8} + \overbrace{9} + \overbrace{7} = 34 \quad \text{worse = high}$$

k=n

$$\overbrace{\begin{array}{cccc} 10 & 8 & 9 & 7 \\ \parallel & \parallel & \parallel & \parallel \\ A & B & C & D \end{array}} = 10 = \text{best}$$

$$\frac{10+31}{2} = \frac{41}{2} = 22$$

K=2

$$\left[\begin{array}{ccccccc} 10 & \dots & 21 & \dots & 34 \end{array} \right]$$

Ask each painter to paint $10 + 22$ units.

$$p_1 \rightarrow 10 + 8 + 9 = 27 \quad (2)$$

$$p_2 \rightarrow 7$$

~~ask 22~~
each painter

$$\left[\begin{array}{c|cc} 10 & 21 \\ & m=15 \end{array} \right] \quad p_1 \rightarrow 10 + 8 + 9 = 27 \quad (2) \quad \text{painters}$$

$$p_2 \rightarrow 7$$

$$\left[\begin{array}{c|cc} 10 & 14 \\ & \text{mid } 12 \end{array} \right] \quad p_1 \rightarrow 10 + 16 = 26 \quad (2)$$

$$p_2 \rightarrow 16$$

$$\left[\begin{array}{c|cc} 10 & 11 \\ & m=10 \end{array} \right] \quad p_1 \rightarrow 10 + 10 = 20 \quad (1)$$

$$p_2 \rightarrow 11$$

$$\left[\begin{array}{c|cc} 10 & 8 & 9 & 7 \\ & p_1 \rightarrow 10 + 8 + 9 = 27 & p_2 \rightarrow 7 & p_3 \rightarrow 7 \end{array} \right] \quad (3)$$

$$\left[\begin{array}{c|cc} 10 & 8 & 9 & 7 \\ & p_1 \rightarrow 10 + 8 = 18 & p_2 \rightarrow 9 + 7 = 16 & \end{array} \right] \quad (2)$$

$$\cancel{b=2}$$

10 10 10 10



Complicated work with 2 pointers

Completed work
with 2 pointers



$$\text{mid} = \cancel{17}$$

$$\left. \begin{array}{l} p_1 \rightarrow 10 \\ p_2 \rightarrow 10 \\ p_3 \rightarrow 10 \\ p_4 \rightarrow 10 \end{array} \right\} \quad \text{4 points}$$

4 points ss



2021

10 10 10 10

$$p_1 \rightarrow \underline{10+10} \quad p_2 \rightarrow \underline{10+10}$$



20

$$\begin{array}{l} p_1 \rightarrow 10 \\ p_2 \rightarrow 10 \\ p_3 \rightarrow 10 \end{array} \quad \left. \begin{array}{l} \downarrow \\ \downarrow \end{array} \right\} 4$$

$$\left. \begin{array}{c} 10 \\ 10 \\ 10+10 \\ 10+10 \end{array} \right\} \text{2 groups}$$

```

5   public static boolean isPossibleToPaint(int [] A, int k,int limit){      //limit is just mid
6     int countOfPainter = 1;
7     int sum = 0;
8
9     for(int ele : A){
10       if(sum + ele <= limit){
11         sum += ele;
12       }
13       else{
14         countOfPainter++;
15         sum = ele;
16       }
17     }
18   }
19   return countOfPainter<=k;
20 }
21

```

```

public static void main(String[] args) {
    Scanner scn = new Scanner(System.in);
    int n = scn.nextInt();
    int [] A= new int[n];
    for(int i = 0; i < n; i++){
        A[i] = scn.nextInt();
    }
    int k = scn.nextInt();      //given painters
    int low = 0;                //max of all elements
    int high = 0;               //sum of all elements
    for(int ele : A){
        low = Math.max(low, ele);
        high += ele;
    }
    int ans = -1;
    while(low <= high){
        int mid = low + (high - low)/2;
        if(isPossibleToPaint(A, k, mid)){
            ans = mid;
            high = mid - 1;
        }
        else{
            low = mid + 1;
        }
    }
    System.out.println(ans);
}

```