10 20 30 40

Pointer-1 — Pointer-2

Jobless

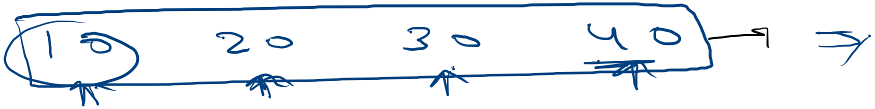
P-3 (P2)

P-1	P-2
10	20 30 40
10 20	30 40
0	10 20 30 40
10 20 30	40

P-2

P-2

mig



while (s <= end)

s = 0 , e = 100

60

30

int Paints = ~~1~~ 2

$10 + 20 < = 75$

int Pbc =

P2

P1 → 10 , 20 , 30

P2 → 40

P3

$\frac{60 \times 2}{120}$

60 Ans

s	e	m
0	100	50
51	100	75
51	74	62
51	61	56
57	61	59
60	61	60
60	59	

ans = ~~75~~ ~~62~~

60 59

51

$\frac{157}{2}$

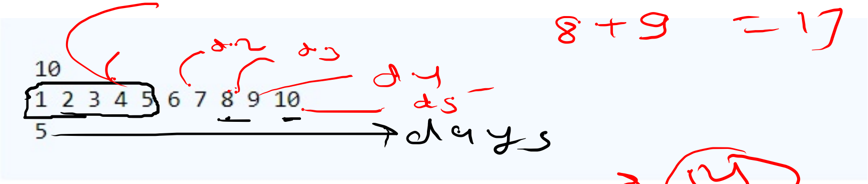
$\frac{51 + 61}{122}$

62

~~$\frac{123}{2}$~~

$\frac{51 + 61}{112}$

$\frac{119}{2}$



```
10
1 2 3 4 5 6 7 8 9 10
5
```

d1 d2 d3 d4 d5 days

8 + 9 = 17

l = ans 27 ~~55~~ ~~2~~

142 = 15

ans = ~~27~~ ~~20~~ ~~16~~ 15

20 — 19 → 15 → 14

P-1 1 2 3 4 5

P-2 → 6, 7

P-3 → 8

P4 → 9

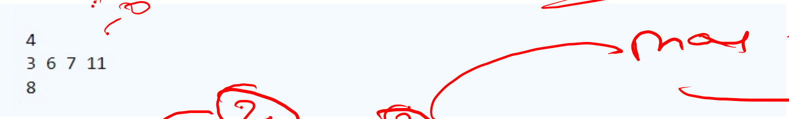
P5 → 10

s	e	m
0	55	27
0	26	13
14	26	20
14	19	16
14	15	14
15	15	15

15 14

15

~ Akash — ms



```
4
3 6 7 11
8
```

may or may not be sorted

3 6 7 11 3

2 2

n=4

max limit = 8

1

3 = 8

2 1

3 8

$\frac{3+5}{2} = 4$

s = 0 , e = 11

mid = 5

$\frac{0+11}{2} = 5.5$

27

0

$\frac{0+5}{}$

$\frac{3+3}{}$

4+3

3.5

$\frac{~~4+~~}{2}$ 13

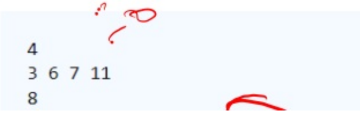
12

5

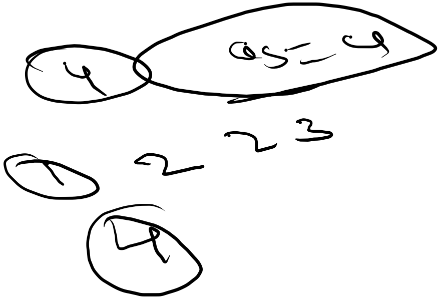
$\frac{0+6}{2}$ 6

→ 12

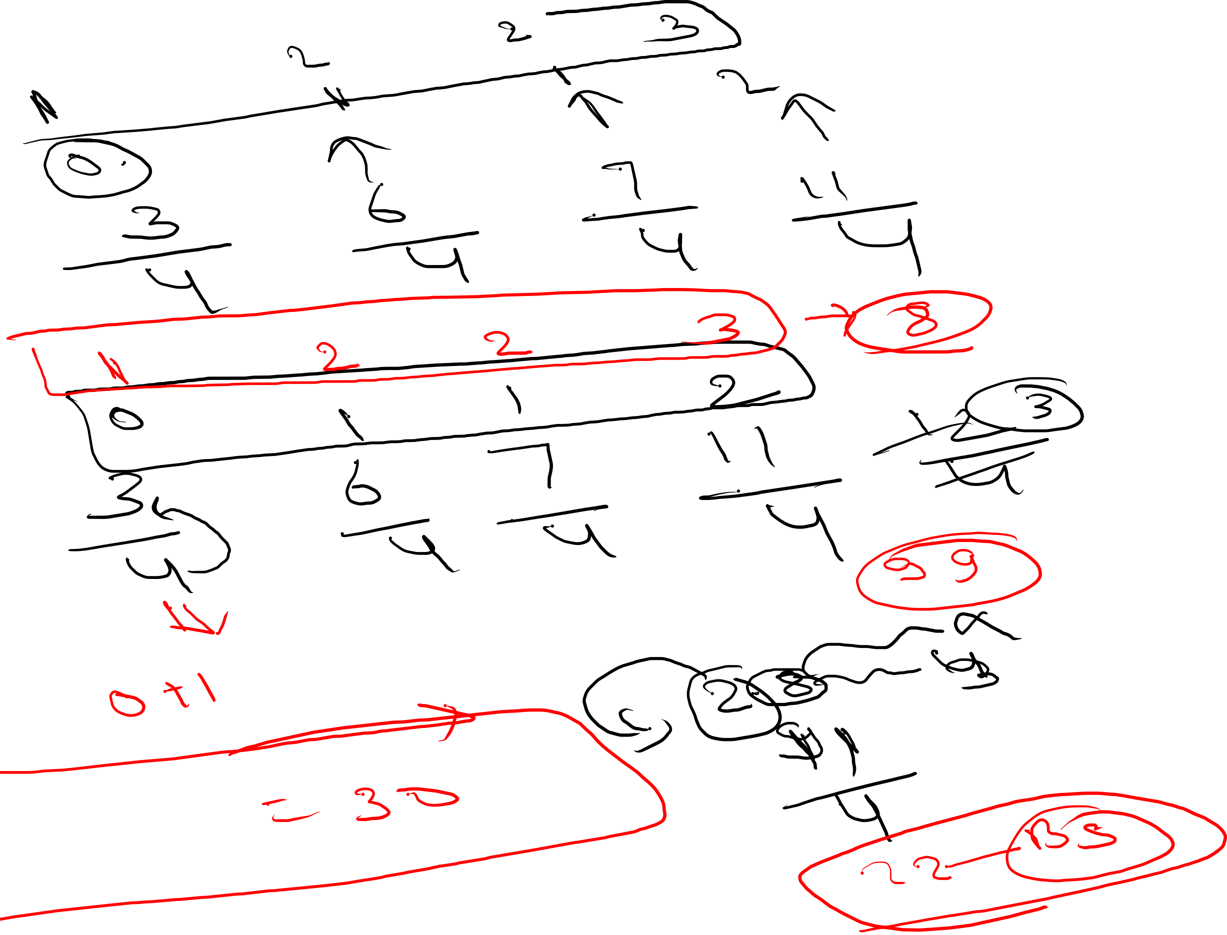
ans = ~~8~~ 4



4
3 6 7 11
8

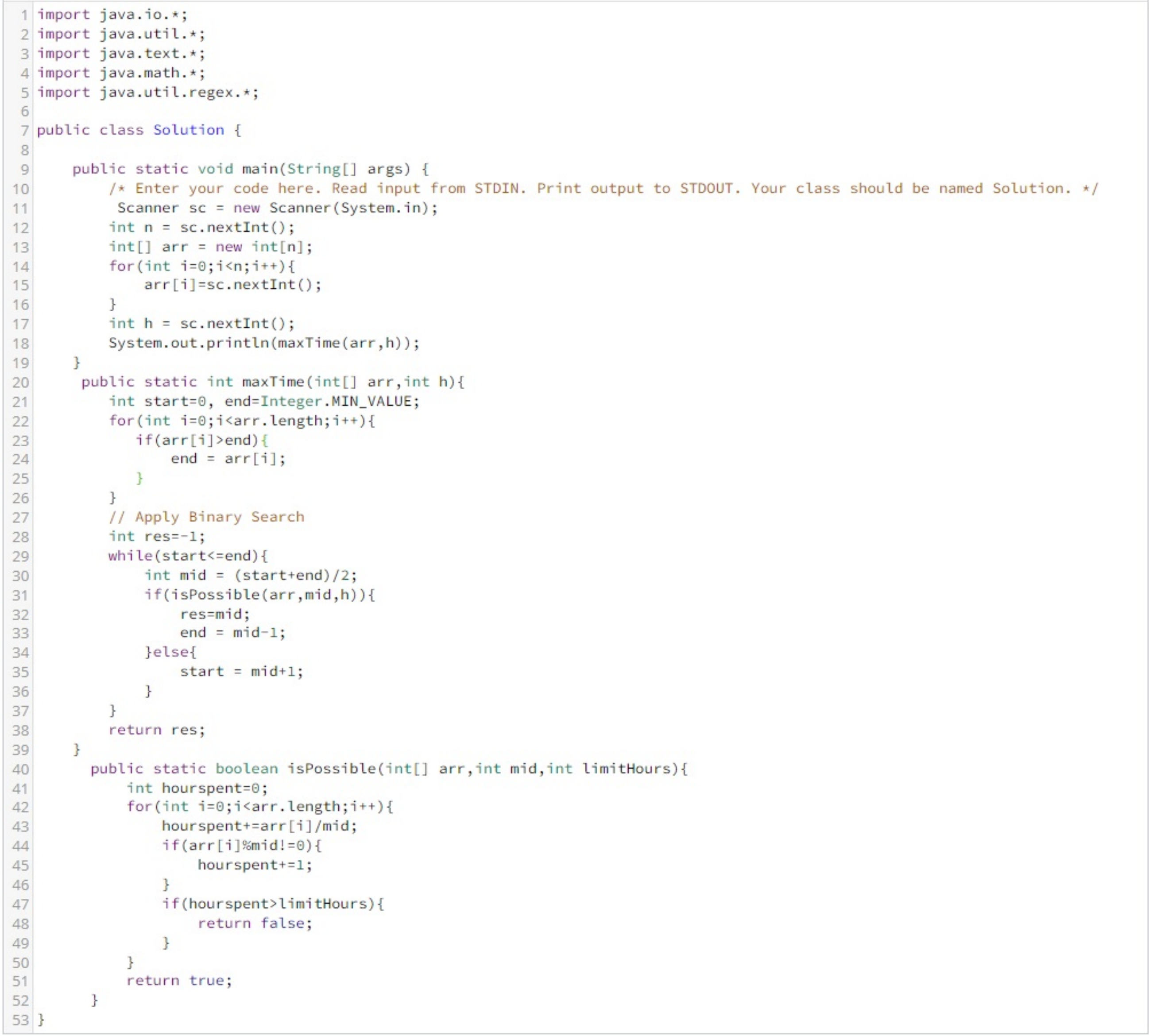


4
ans = 4
1 2 2 3
4



N 2 2 3
0
3/4
6/4
7/4
11/4
8
N 2 2 3
0 1 1 2
3/4 6/4 7/4 11/4
3
99
0+1
= 30
28
22
135

The banana challenge

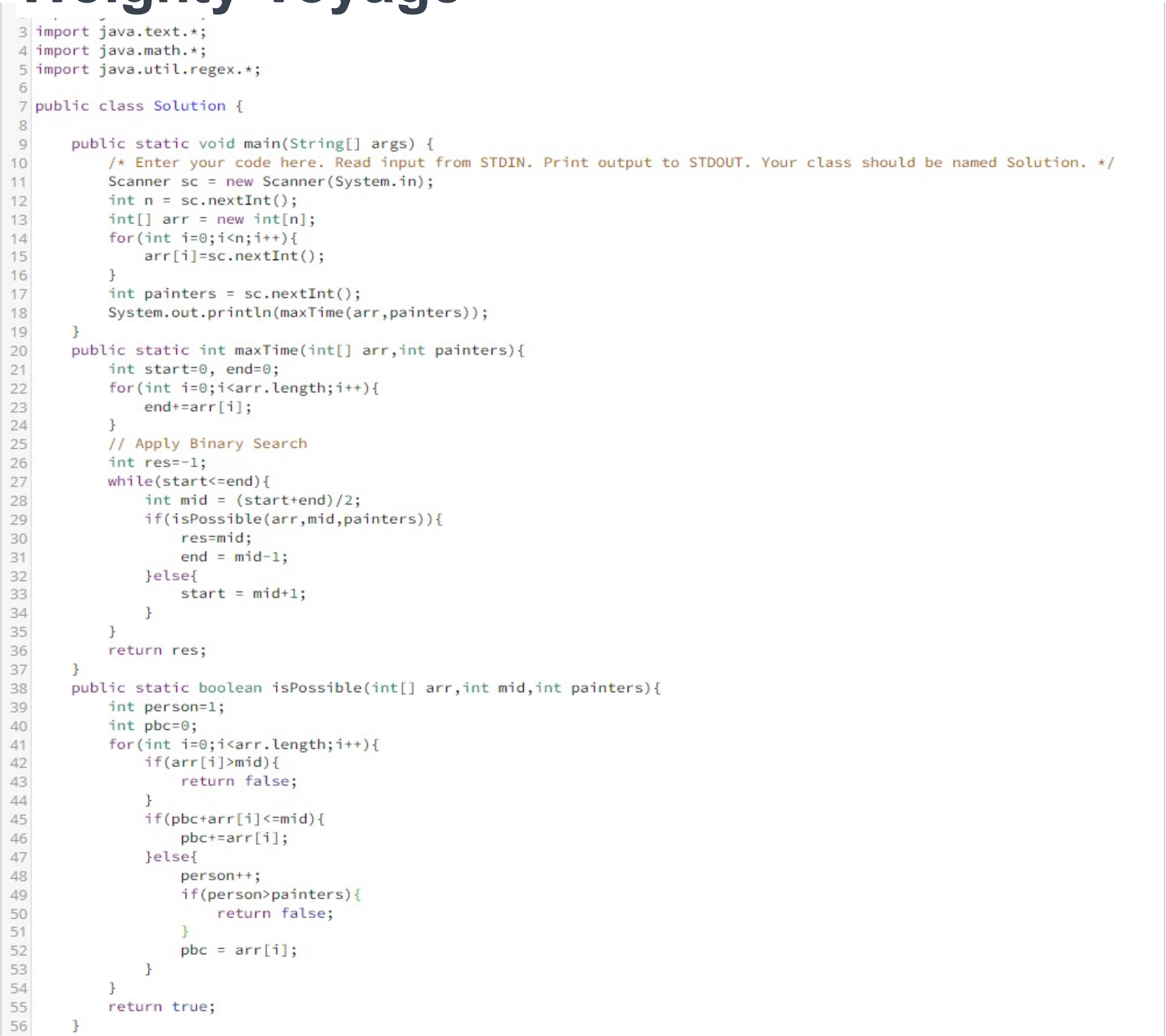


```
import java.io.*;
import java.util.*;
import java.text.*;
import java.math.*;
import java.util.regex.*;

public class Solution {

    public static void main(String[] args) {
        /* Enter your code here. Read input from STDIN. Print output to STDOUT. Your class should be named Solution. */
         Scanner sc = new Scanner(System.in);
        int n = sc.nextInt();
        int[] arr = new int[n];
        for(int i=0;i<n;i++){
            arr[i]=sc.nextInt();
        }
        int h = sc.nextInt();
        System.out.println(maxTime(arr,h));
    }
     public static int maxTime(int[] arr,int h){
        int start=0, end=Integer.MIN_VALUE;
        for(int i=0;i<arr.length;i++){
           if(arr[i]>end){
               end = arr[i];
           }
        }
        // Apply Binary Search
        int res=-1;
        while(start<=end){
            int mid = (start+end)/2;
            if(isPossible(arr,mid,h)){
                res=mid;
                end = mid-1;
            }else{
                start = mid+1;
            }
        }
        return res;
    }
      public static boolean isPossible(int[] arr,int mid,int limitHours){
          int hourspent=0;
          for(int i=0;i<arr.length;i++){
              hourspent+=arr[i]/mid;
              if(arr[i]%mid!=0){
                  hourspent+=1;
              }
              if(hourspent>limitHours){
                  return false;
              }
          }
          return true;
      }
}
```

Weighty Voyage

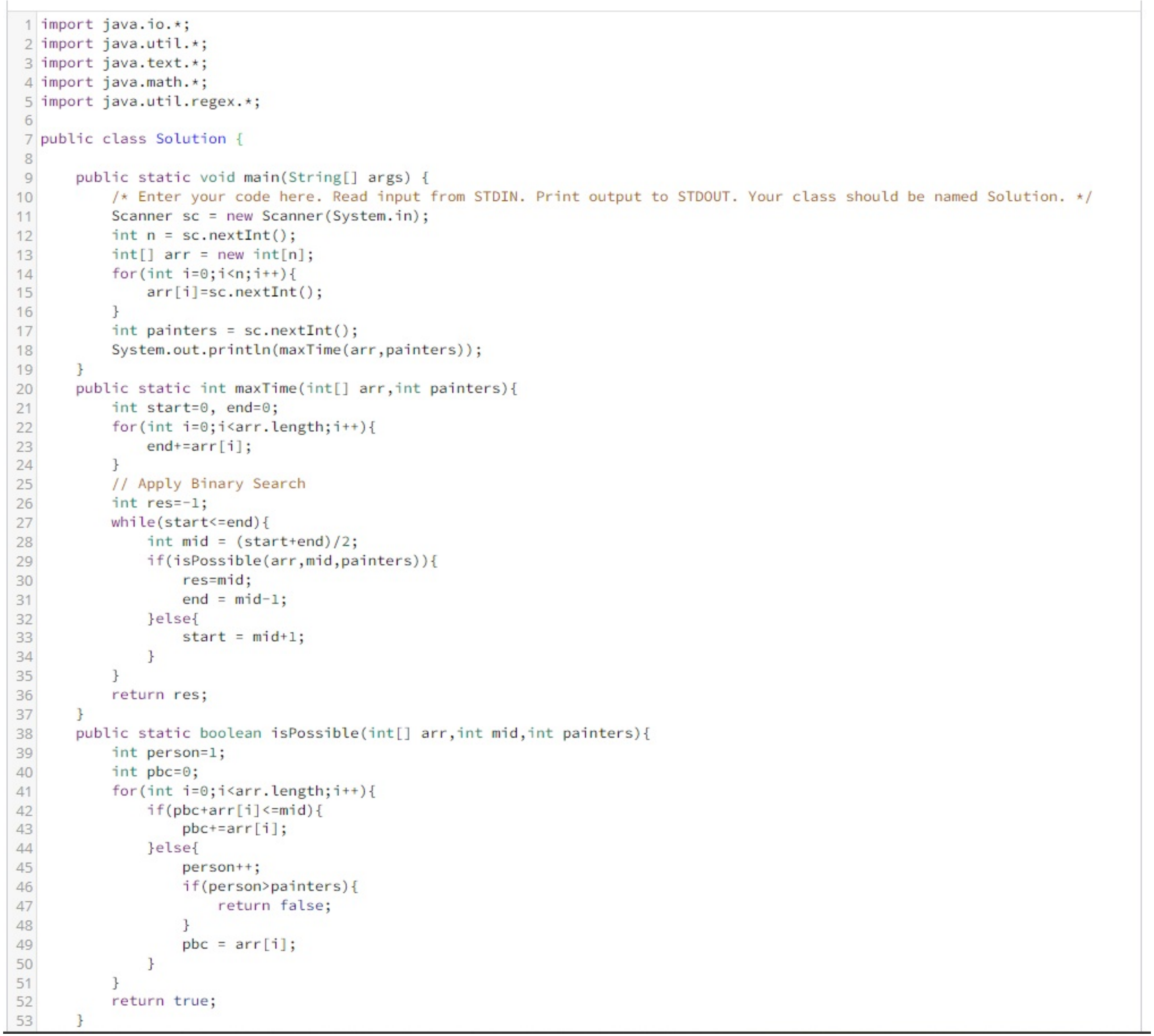


```java
import java.text.*;
import java.math.*;
import java.util.regex.*;

public class Solution {

    public static void main(String[] args) {
        /* Enter your code here. Read input from STDIN. Print output to STDOUT. Your class should be named Solution. */
        Scanner sc = new Scanner(System.in);
        int n = sc.nextInt();
        int[] arr = new int[n];
        for(int i=0;i<n;i++){
            arr[i]=sc.nextInt();
        }
        int painters = sc.nextInt();
        System.out.println(maxTime(arr,painters));
    }
    public static int maxTime(int[] arr,int painters){
        int start=0, end=0;
        for(int i=0;i<arr.length;i++){
            end+=arr[i];
        }
        // Apply Binary Search
        int res=-1;
        while(start<=end){
            int mid = (start+end)/2;
            if(isPossible(arr,mid,painters)){
                res=mid;
                end = mid-1;
            }else{
                start = mid+1;
            }
        }
        return res;
    }
    public static boolean isPossible(int[] arr,int mid,int painters){
        int person=1;
        int pbc=0;
        for(int i=0;i<arr.length;i++){
            if(arr[i]>mid){
                return false;
            }
            if(pbc+arr[i]<=mid){
                pbc+=arr[i];
            }else{
                person++;
                if(person>painters){
                    return false;
                }
                pbc = arr[i];
            }
        }
        return true;
    }
```

The painter



```java
import java.io.*;
import java.util.*;
import java.text.*;
import java.math.*;
import java.util.regex.*;

public class Solution {

    public static void main(String[] args) {
        /* Enter your code here. Read input from STDIN. Print output to STDOUT. Your class should be named Solution. */
        Scanner sc = new Scanner(System.in);
        int n = sc.nextInt();
        int[] arr = new int[n];
        for(int i=0;i<n;i++){
            arr[i]=sc.nextInt();
        }
        int painters = sc.nextInt();
        System.out.println(maxTime(arr,painters));
    }
    public static int maxTime(int[] arr,int painters){
        int start=0, end=0;
        for(int i=0;i<arr.length;i++){
            end+=arr[i];
        }
        // Apply Binary Search
        int res=-1;
        while(start<=end){
            int mid = (start+end)/2;
            if(isPossible(arr,mid,painters)){
                res=mid;
                end = mid-1;
            }else{
                start = mid+1;
            }
        }
        return res;
    }
    public static boolean isPossible(int[] arr,int mid,int painters){
        int person=1;
        int pbc=0;
        for(int i=0;i<arr.length;i++){
            if(pbc+arr[i]<=mid){
                pbc+=arr[i];
            }else{
                person++;
                if(person>painters){
                    return false;
                }
                pbc = arr[i];
            }
        }
        return true;
    }
```