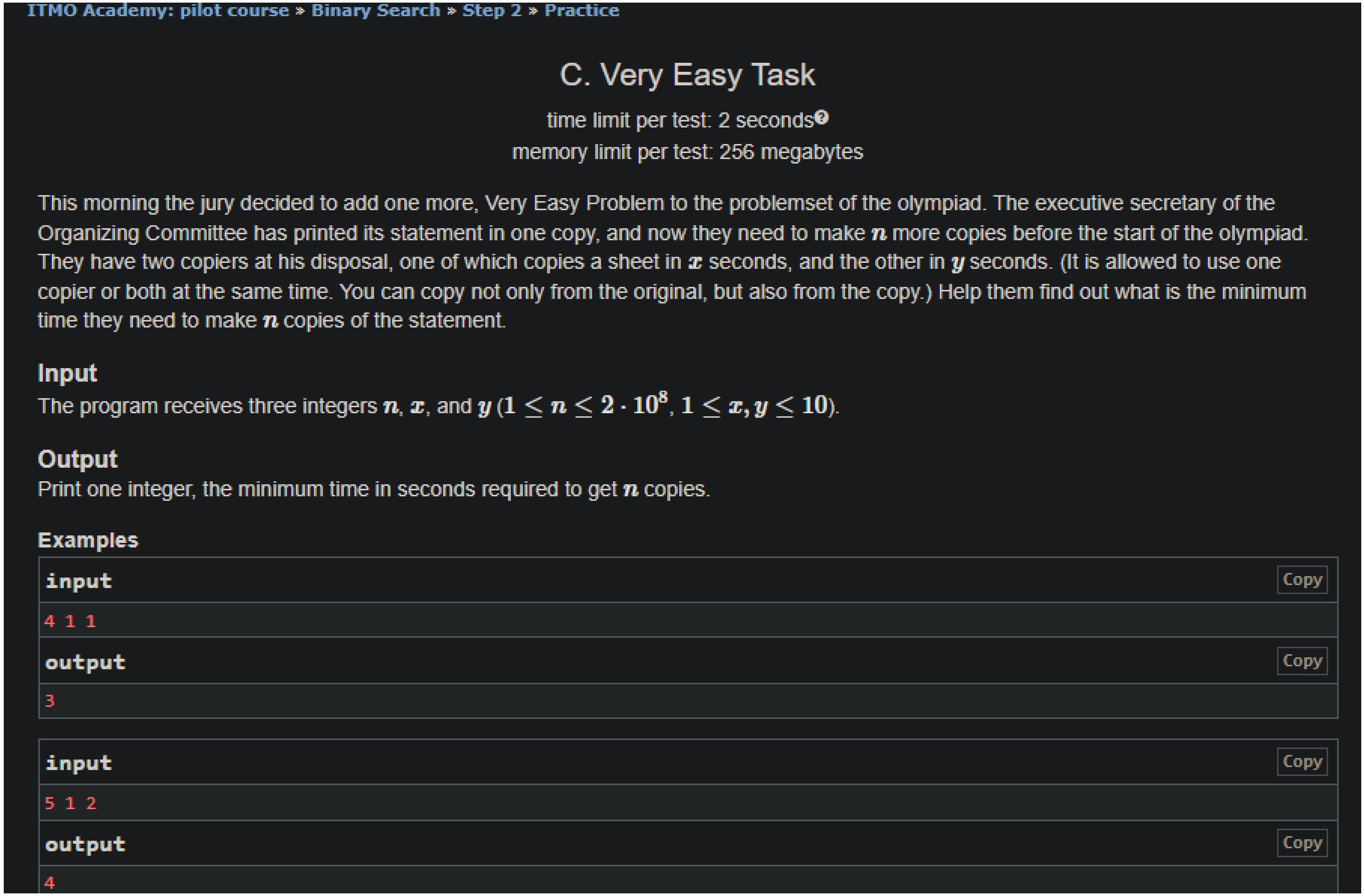


ITMO Academy: pilot course » Binary Search » Step 2 » Practice

C. Very Easy Task

time limit per test: 2 seconds

memory limit per test: 256 megabytes

This morning the jury decided to add one more, Very Easy Problem to the problemset of the olympiad. The executive secretary of the Organizing Committee has printed its statement in one copy, and now they need to make n more copies before the start of the olympiad. They have two copiers at his disposal, one of which copies a sheet in x seconds, and the other in y seconds. (It is allowed to use one copier or both at the same time. You can copy not only from the original, but also from the copy.) Help them find out what is the minimum time they need to make n copies of the statement.

Input

The program receives three integers n, x, and y ($1 \le n \le 2 \cdot 10^8$, $1 \le x, y \le 10$).

Output

Print one integer, the minimum time in seconds required to get n copies.

Examples

input

```
4 1 1
```

output

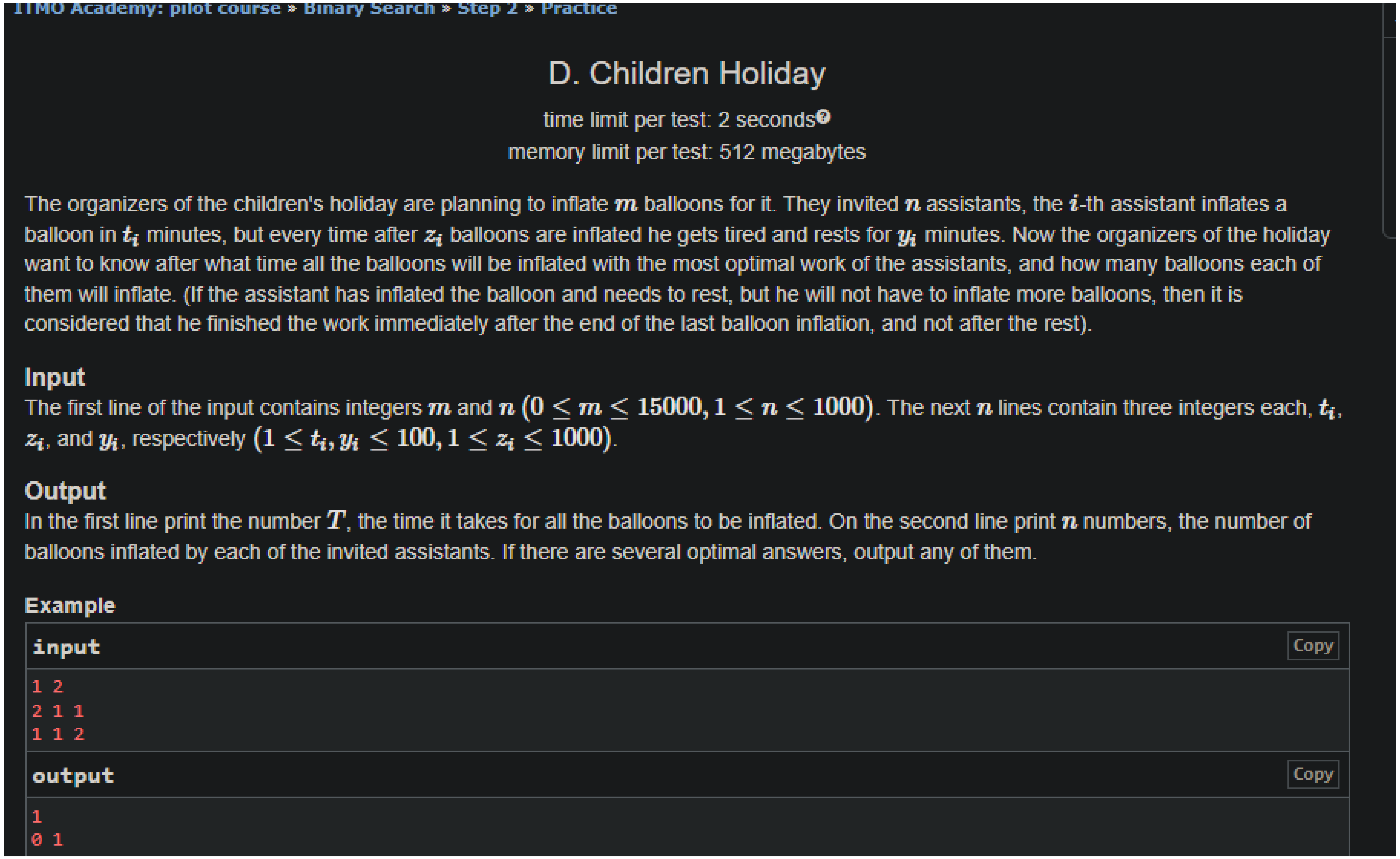
```
3
```

input

```
5 1 2
```

output

```
4
```



D. Children Holiday

time limit per test: 2 seconds

memory limit per test: 512 megabytes

The organizers of the children's holiday are planning to inflate m balloons for it. They invited n assistants, the i-th assistant inflates a balloon in t_i minutes, but every time after z_i balloons are inflated he gets tired and rests for y_i minutes. Now the organizers of the holiday want to know after what time all the balloons will be inflated with the most optimal work of the assistants, and how many balloons each of them will inflate. (If the assistant has inflated the balloon and needs to rest, but he will not have to inflate more balloons, then it is considered that he finished the work immediately after the end of the last balloon inflation, and not after the rest).

Input

The first line of the input contains integers m and n $(0 \le m \le 15000, 1 \le n \le 1000)$. The next n lines contain three integers each, t_i, z_i, and y_i, respectively $(1 \le t_i, y_i \le 100, 1 \le z_i \le 1000)$.

Output

In the first line print the number T, the time it takes for all the balloons to be inflated. On the second line print n numbers, the number of balloons inflated by each of the invited assistants. If there are several optimal answers, output any of them.

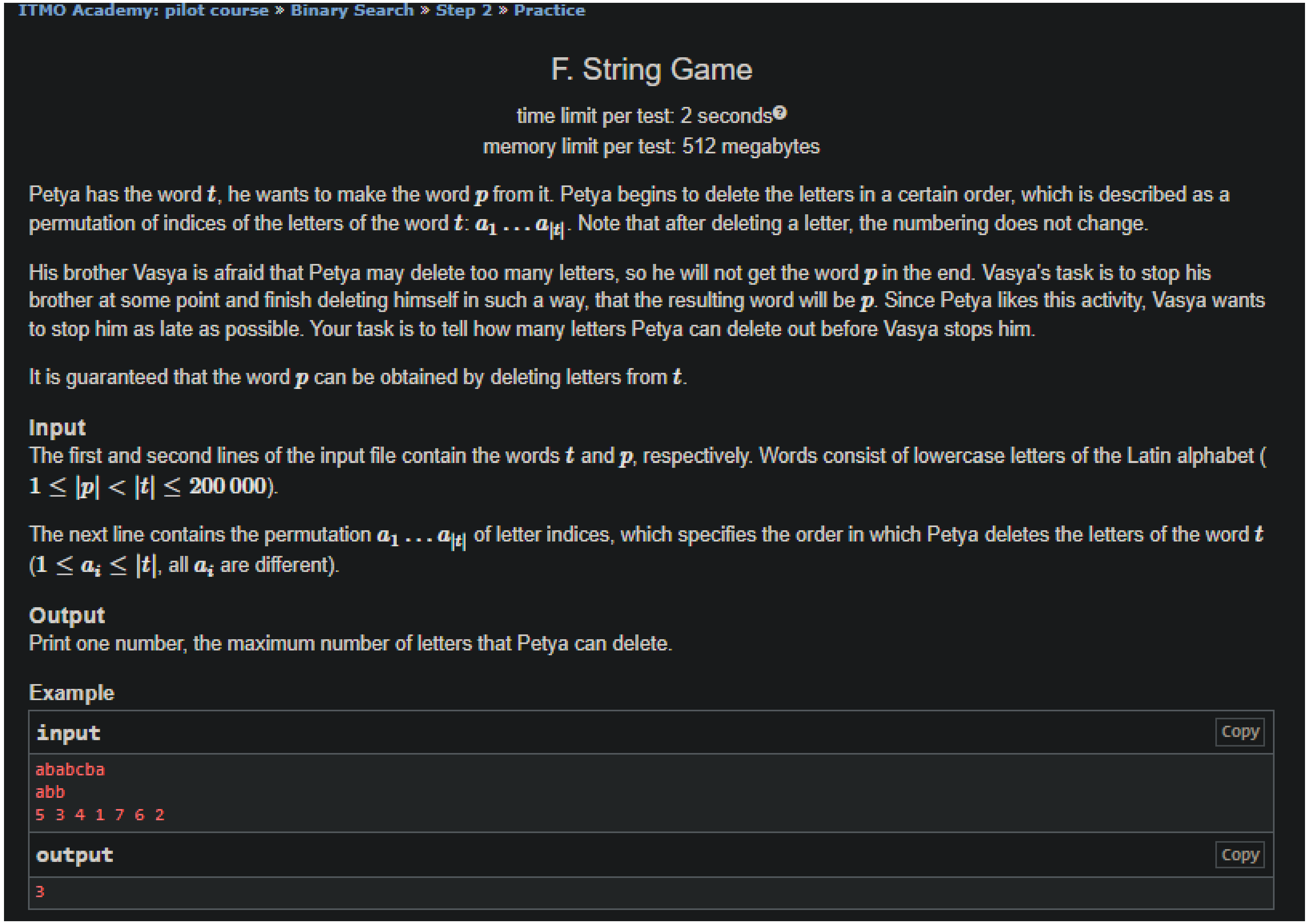
Example

input Copy

```
1 2
2 1 1
1 1 2
```

output Copy

```
1
0 1
```


ITMO Academy: pilot course » Binary Search » Step 2 » Practice

F. String Game

time limit per test: 2 seconds
memory limit per test: 512 megabytes

Petya has the word t, he wants to make the word p from it. Petya begins to delete the letters in a certain order, which is described as a permutation of indices of the letters of the word t: $a_1 \dots a_{|t|}$. Note that after deleting a letter, the numbering does not change.

His brother Vasya is afraid that Petya may delete too many letters, so he will not get the word p in the end. Vasya's task is to stop his brother at some point and finish deleting himself in such a way, that the resulting word will be p. Since Petya likes this activity, Vasya wants to stop him as late as possible. Your task is to tell how many letters Petya can delete out before Vasya stops him.

It is guaranteed that the word p can be obtained by deleting letters from t.

Input

The first and second lines of the input file contain the words t and p, respectively. Words consist of lowercase letters of the Latin alphabet ($1 \le |p| < |t| \le 200\,000$).

The next line contains the permutation $a_1 \dots a_{|t|}$ of letter indices, which specifies the order in which Petya deletes the letters of the word t ($1 \le a_i \le |t|$, all a_i are different).

Output

Print one number, the maximum number of letters that Petya can delete.

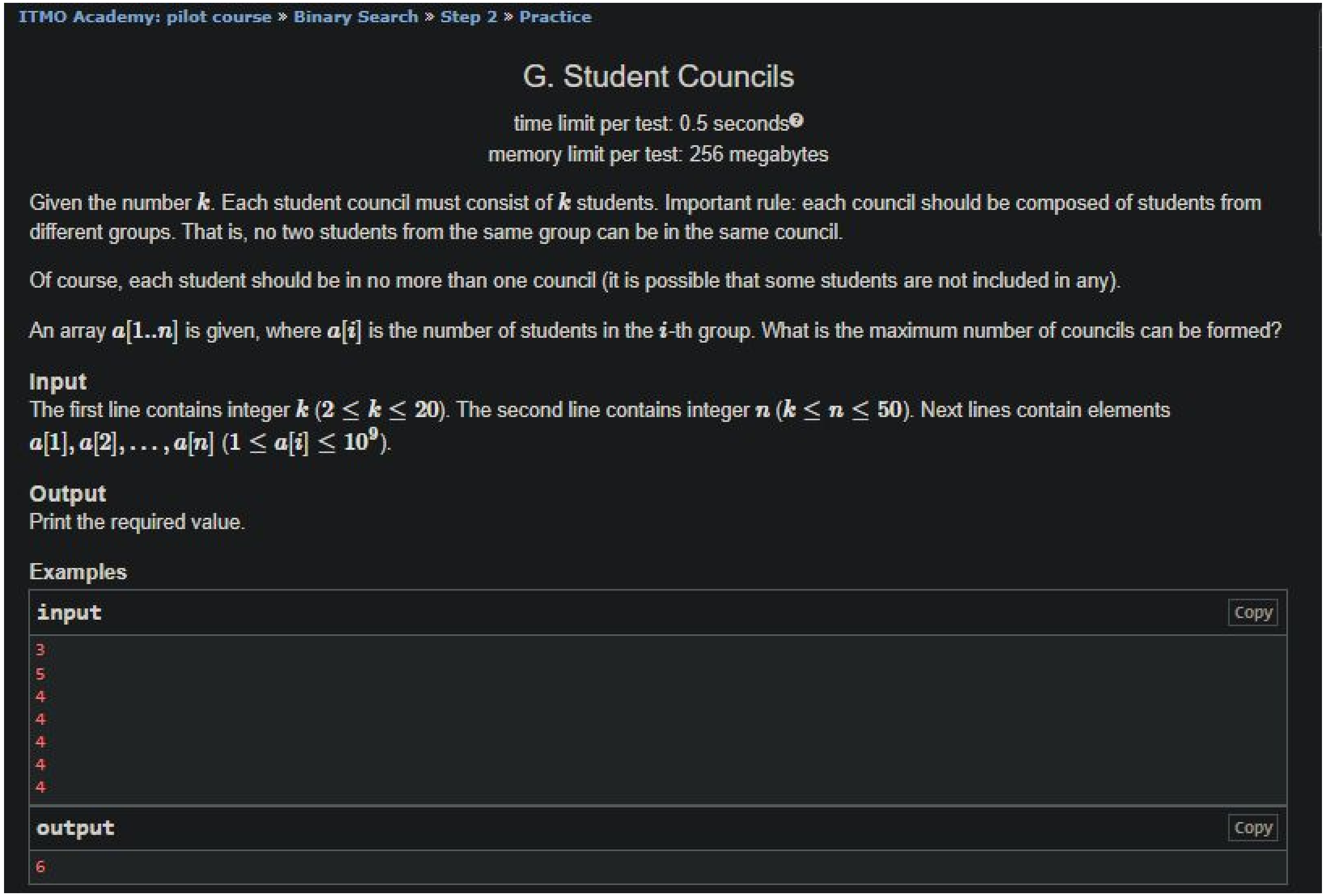
Example

input

```
ababcba
abb
5 3 4 1 7 6 2
```

output

```
3
```


ITMO Academy: pilot course » Binary Search » Step 2 » Practice

G. Student Councils

time limit per test: 0.5 seconds
memory limit per test: 256 megabytes

Given the number k. Each student council must consist of k students. Important rule: each council should be composed of students from different groups. That is, no two students from the same group can be in the same council.

Of course, each student should be in no more than one council (it is possible that some students are not included in any).

An array $a[1..n]$ is given, where $a[i]$ is the number of students in the i-th group. What is the maximum number of councils can be formed?

Input
The first line contains integer k $(2 \le k \le 20)$. The second line contains integer n $(k \le n \le 50)$. Next lines contain elements $a[1], a[2], \ldots, a[n]$ $(1 \le a[i] \le 10^9)$.

Output
Print the required value.

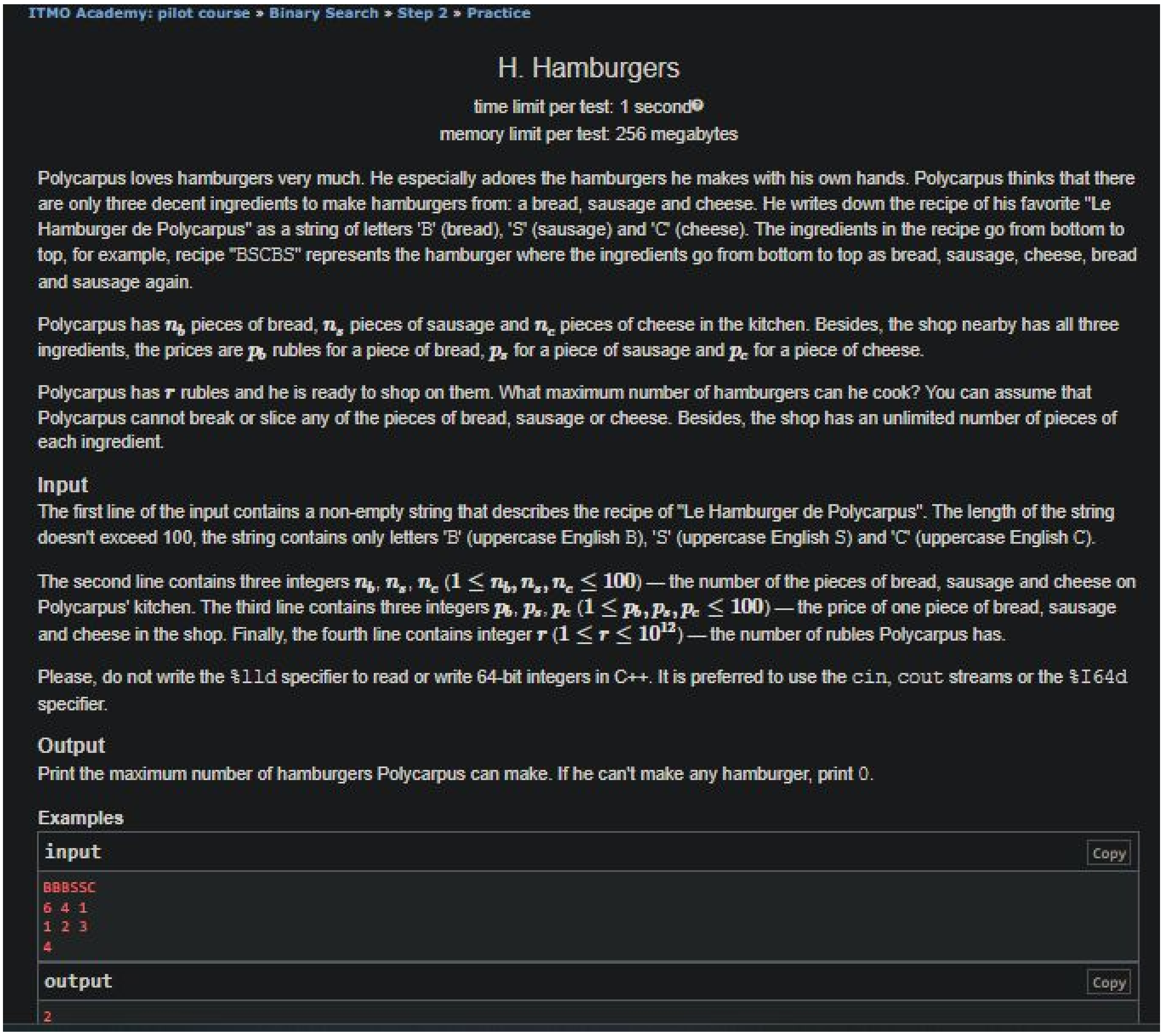
Examples

input

```
3
5
4
4
4
4
4
```

output

```
6
```



H. Hamburgers

time limit per test: 1 second
memory limit per test: 256 megabytes

Polycarpus loves hamburgers very much. He especially adores the hamburgers he makes with his own hands. Polycarpus thinks that there are only three decent ingredients to make hamburgers from: a bread, sausage and cheese. He writes down the recipe of his favorite "Le Hamburger de Polycarpus" as a string of letters 'B' (bread), 'S' (sausage) and 'C' (cheese). The ingredients in the recipe go from bottom to top, for example, recipe "BSCBS" represents the hamburger where the ingredients go from bottom to top as bread, sausage, cheese, bread and sausage again.

Polycarpus has n_b pieces of bread, n_s pieces of sausage and n_c pieces of cheese in the kitchen. Besides, the shop nearby has all three ingredients, the prices are p_b rubles for a piece of bread, p_s for a piece of sausage and p_c for a piece of cheese.

Polycarpus has r rubles and he is ready to shop on them. What maximum number of hamburgers can he cook? You can assume that Polycarpus cannot break or slice any of the pieces of bread, sausage or cheese. Besides, the shop has an unlimited number of pieces of each ingredient.

Input

The first line of the input contains a non-empty string that describes the recipe of "Le Hamburger de Polycarpus". The length of the string doesn't exceed 100, the string contains only letters 'B' (uppercase English B), 'S' (uppercase English S) and 'C' (uppercase English C).

The second line contains three integers n_b, n_s, n_c ($1 \leq n_b, n_s, n_c \leq 100$) — the number of the pieces of bread, sausage and cheese on Polycarpus' kitchen. The third line contains three integers p_b, p_s, p_c ($1 \leq p_b, p_s, p_c \leq 100$) — the price of one piece of bread, sausage and cheese in the shop. Finally, the fourth line contains integer r ($1 \leq r \leq 10^{12}$) — the number of rubles Polycarpus has.

Please, do not write the `%lld` specifier to read or write 64-bit integers in C++. It is preferred to use the `cin`, `cout` streams or the `%I64d` specifier.

Output

Print the maximum number of hamburgers Polycarpus can make. If he can't make any hamburger, print 0.

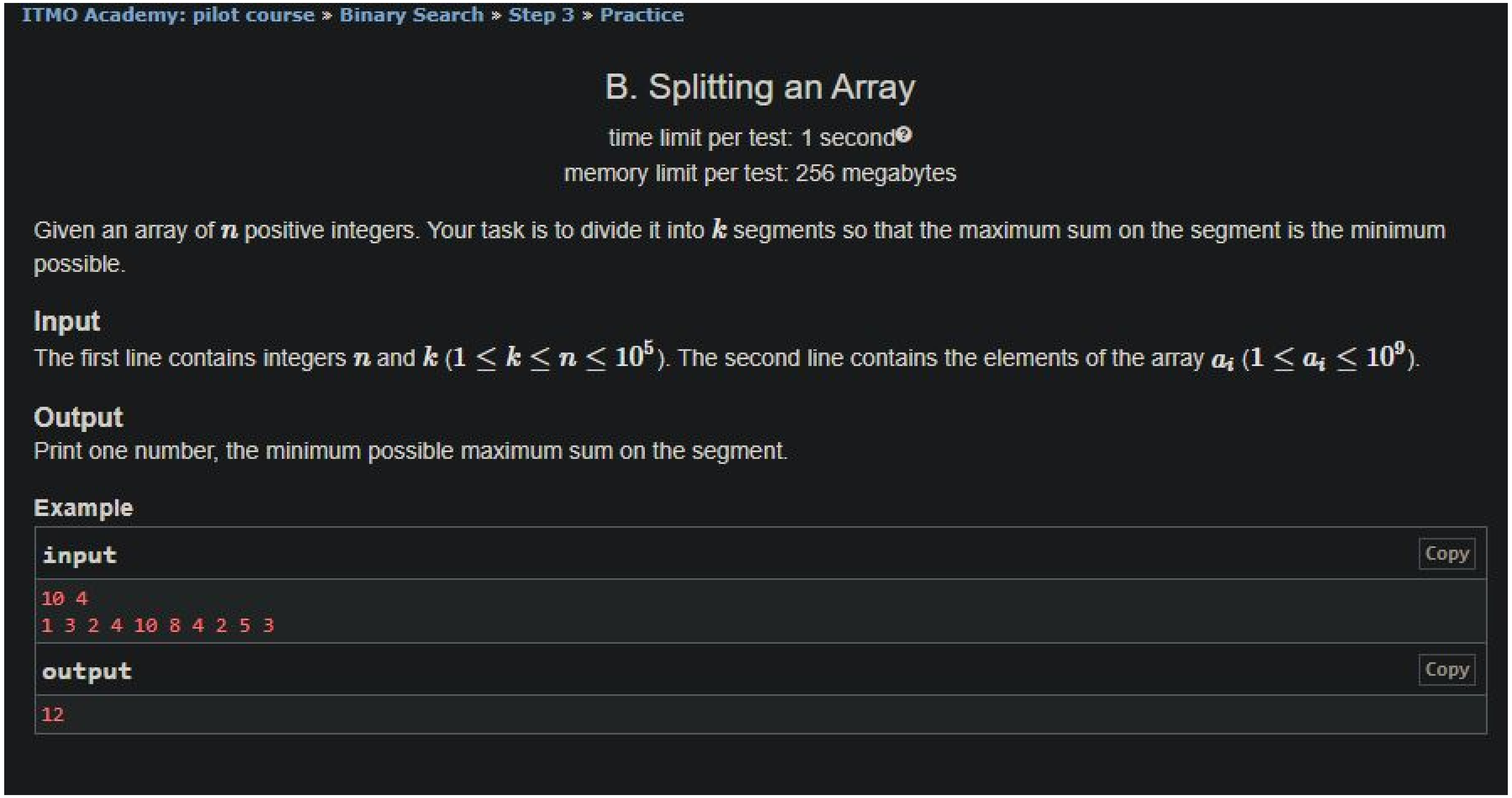
Examples

input

```
BBBSSC
6 4 1
1 2 3
4
```

output

```
2
```



B. Splitting an Array

time limit per test: 1 second

memory limit per test: 256 megabytes

Given an array of n positive integers. Your task is to divide it into k segments so that the maximum sum on the segment is the minimum possible.

Input

The first line contains integers n and k $(1 \le k \le n \le 10^5)$. The second line contains the elements of the array a_i $(1 \le a_i \le 10^9)$.

Output

Print one number, the minimum possible maximum sum on the segment.

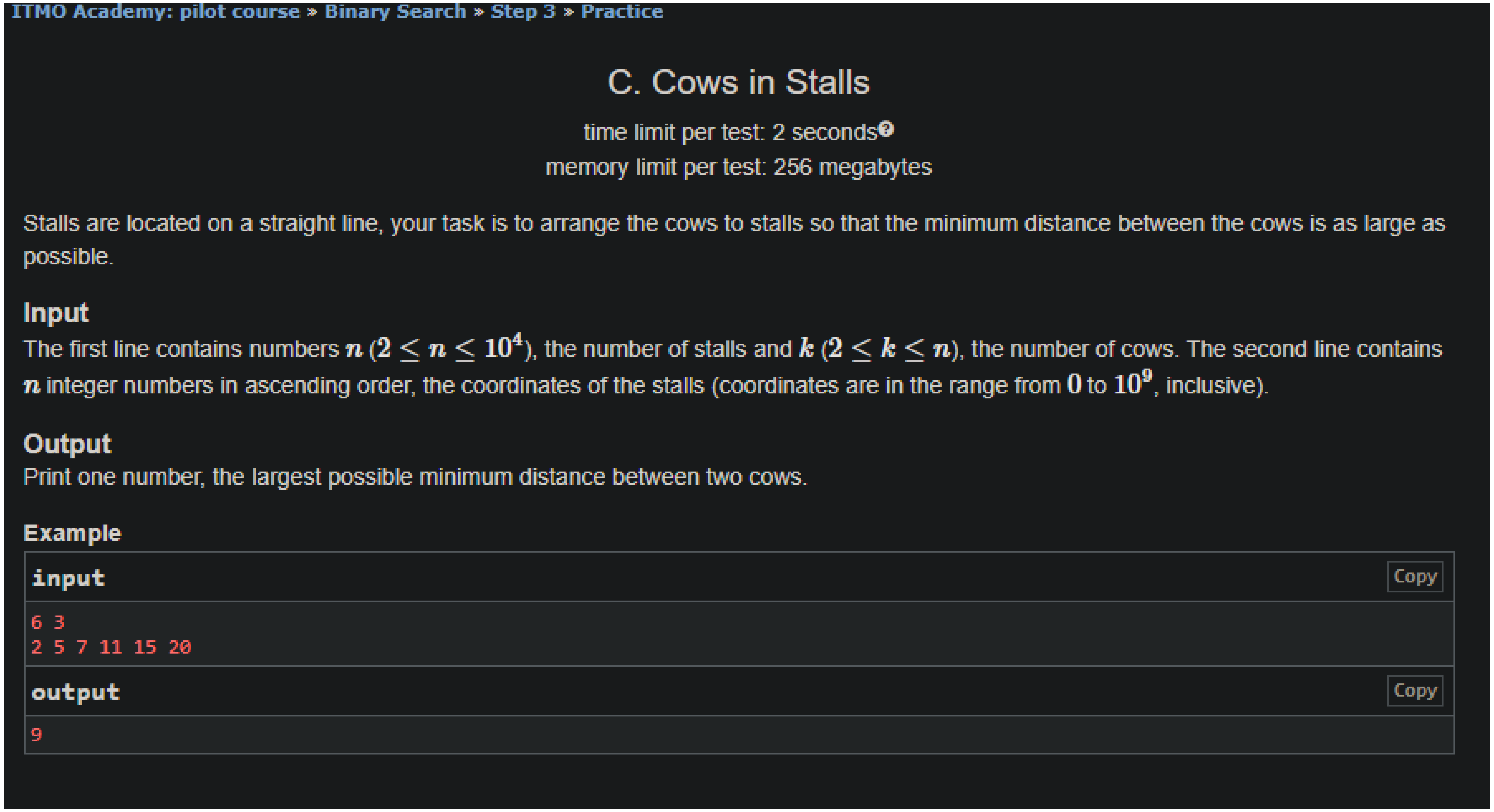
Example

input

```
10 4
1 3 2 4 10 8 4 2 5 3
```

output

```
12
```


ITMO Academy: pilot course » Binary Search » Step 3 » Practice

C. Cows in Stalls

time limit per test: 2 seconds

memory limit per test: 256 megabytes

Stalls are located on a straight line, your task is to arrange the cows to stalls so that the minimum distance between the cows is as large as possible.

Input

The first line contains numbers n ($2 \le n \le 10^4$), the number of stalls and k ($2 \le k \le n$), the number of cows. The second line contains n integer numbers in ascending order, the coordinates of the stalls (coordinates are in the range from 0 to 10^9, inclusive).

Output

Print one number, the largest possible minimum distance between two cows.

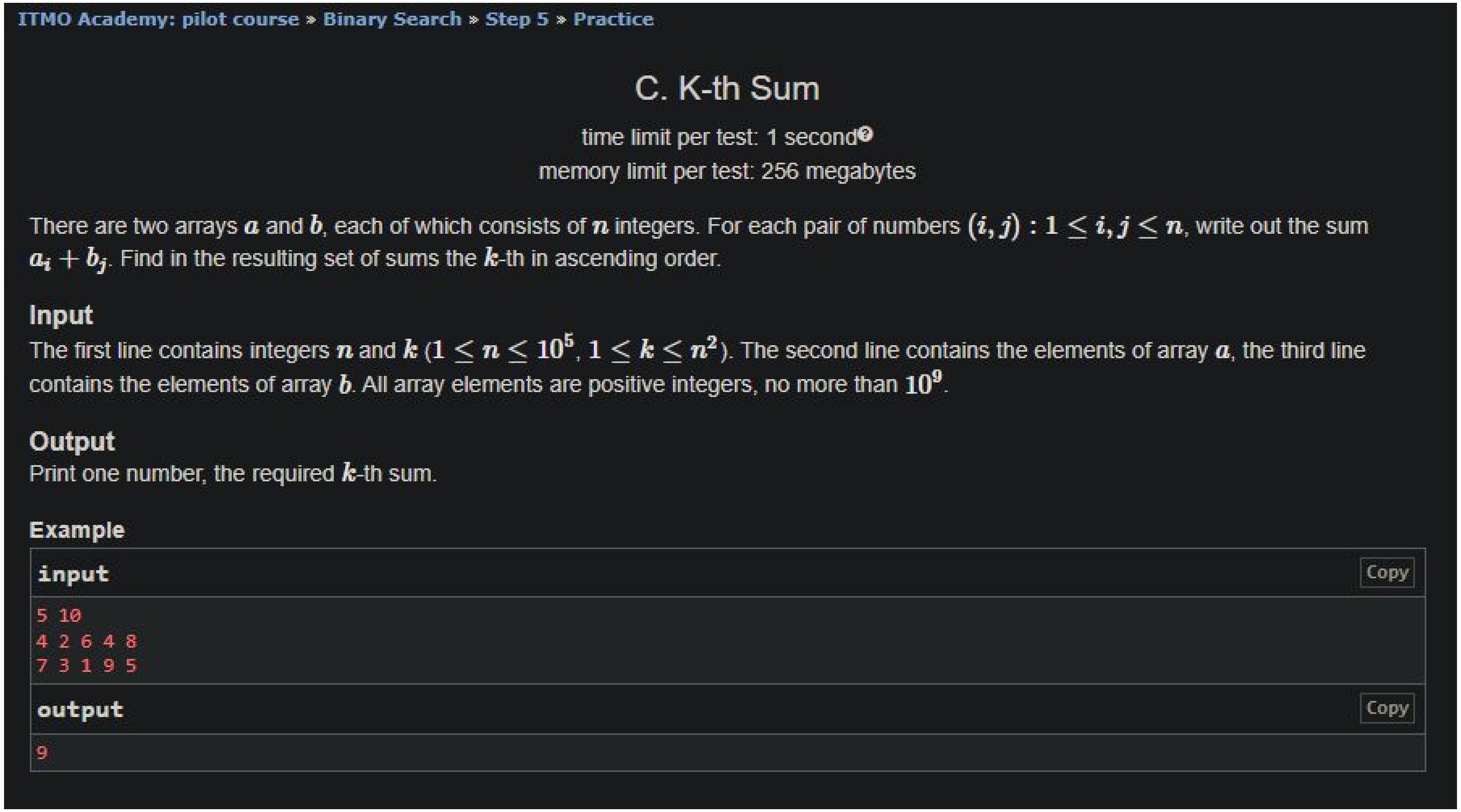
Example

input

```
6 3
2 5 7 11 15 20
```

output

```
9
```


ITMO Academy: pilot course » Binary Search » Step 5 » Practice

C. K-th Sum

time limit per test: 1 second
memory limit per test: 256 megabytes

There are two arrays a and b, each of which consists of n integers. For each pair of numbers $(i, j) : 1 \le i, j \le n$, write out the sum $a_i + b_j$. Find in the resulting set of sums the k-th in ascending order.

Input

The first line contains integers n and k ($1 \le n \le 10^5$, $1 \le k \le n^2$). The second line contains the elements of array a, the third line contains the elements of array b. All array elements are positive integers, no more than 10^9.

Output

Print one number, the required k-th sum.

Example

input

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

output

```
9
```

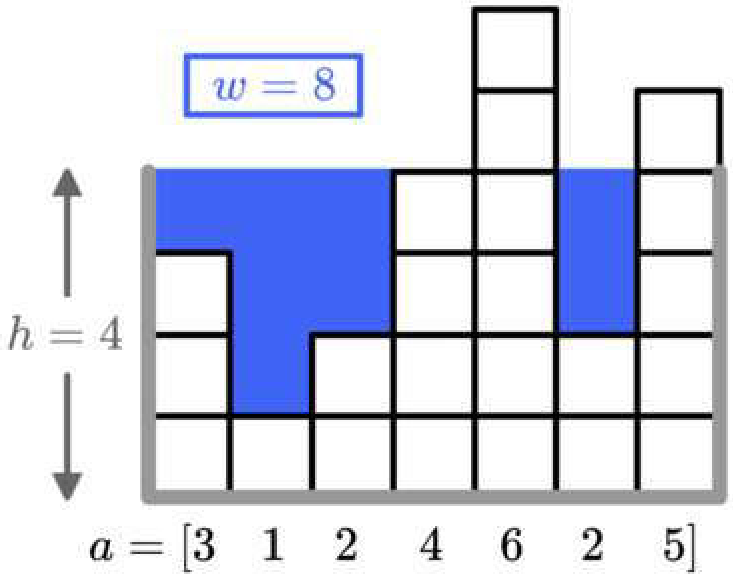
Problem B. Building an Aquarium

Time limit 2000 ms
Mem limit 262144 kB
Problem Link

You love fish, that's why you have decided to build an aquarium. You have a piece of coral made of n columns, the i-th of which is a_i units tall. Afterwards, you will build a tank around the coral as follows:

- Pick an integer $h \geq 1$ — the *height* of the tank. Build walls of height h on either side of the tank.
- Then, fill the tank up with water so that the height of each column is h, unless the coral is taller than h; then no water should be added to this column.

For example, with $a = [3, 1, 2, 4, 6, 2, 5]$ and a height of $h = 4$, you will end up using a total of $w = 8$ units of water, as shown.



You can use at most x units of water to fill up the tank, but you want to build the biggest tank possible. What is the largest value of h you can select?

Input

The first line contains a single integer t ($1 \leq t \leq 10^4$) — the number of test cases.

The first line of each test case contains two positive integers n and x ($1 \leq n \leq 2 \cdot 10^5$; $1 \leq x \leq 10^9$) — the number of columns of the coral and the maximum amount of water you can use.

The second line of each test case contains n space-separated integers a_i ($1 \leq a_i \leq 10^9$) — the heights of the coral.

The sum of n over all test cases doesn't exceed $2 \cdot 10^5$.

Output

For each test case, output a single positive integer h ($h \geq 1$) — the maximum height the tank can have, so you need at most x units of water to fill up the tank.

We have a proof that under these constraints, such a value of h always exists.

Examples

Input	Output

5
7 9
3 1 2 4 6 2 5
3 10
1 1 1
4 1
1 4 3 4
6 1984
2 6 5 9 1 8
1 1000000000
1
	4
	4
	2
	335
	1000000001

Note

The first test case is pictured in the statement. With $h = 4$ we need 8 units of water, but if h is increased to 5 we need 13 units of water, which is more than $x = 9$. So $h = 4$ is optimal.

In the second test case, we can pick $h = 4$ and add 3 units to each column, using a total of 9 units of water. It can be shown that this is optimal.

In the third test case, we can pick $h = 2$ and use all of our water, so it is optimal.

Problem C. Save More Mice

Time limit 4000 ms
Mem limit 262144 kB
Problem Link

There are one cat, k mice, and one hole on a coordinate line. The cat is located at the point 0, the hole is located at the point n. All mice are located between the cat and the hole: the i-th mouse is located at the point x_i ($0 < x_i < n$). At each point, many mice can be located.

In one second, the following happens. First, **exactly one** mouse moves to the right by 1. If the mouse reaches the hole, it hides (i.e. the mouse will not any more move to any point and will not be eaten by the cat). Then (**after that** the mouse has finished its move) the cat moves to the right by 1. If at the new cat's position, some mice are located, the cat eats them (they will not be able to move after that). The actions are performed until any mouse hasn't been hidden or isn't eaten.

In other words, the first move is made by a mouse. If the mouse has reached the hole, it's saved. Then the cat makes a move. The cat eats the mice located at the pointed the cat has reached (if the cat has reached the hole, it eats nobody).

Each second, you can select a mouse that will make a move. What is the maximum number of mice that can reach the hole without being eaten?

Input

The first line contains one integer t ($1 \le t \le 10^4$) — the number of test cases. Then t test cases follow.

Each test case consists of two lines. The first line contains two integers n and k ($2 \le n \le 10^9$, $1 \le k \le 4 \cdot 10^5$). The second line contains k integers $x_1, x_2, \dots x_k$ ($1 \le x_i < n$) — the initial coordinates of the mice.

It is guaranteed that the sum of all k given in the input doesn't exceed $4 \cdot 10^5$.

Output

For each test case output on a separate line an integer m ($m \geq 0$) — the maximum number of mice that can reach the hole without being eaten.

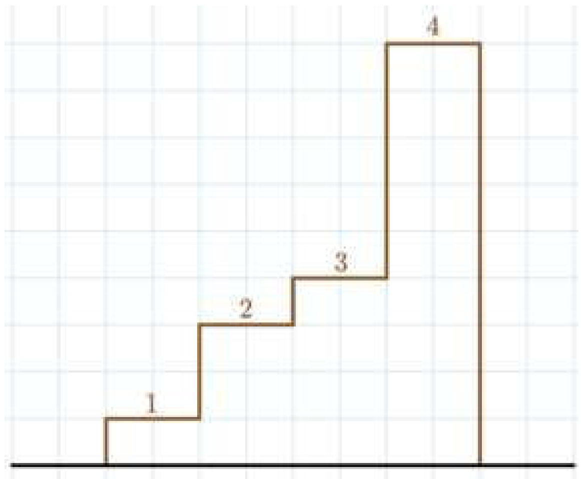
Examples

Input	Output
3	3
10 6	1
8 7 5 4 9 4	4
2 8	
1 1 1 1 1 1 1 1	
12 11	
1 2 3 4 5 6 7 8 9 10 11	

Problem D. Scuza

Time limit 3000 ms
Mem limit 262144 kB
Problem Link

Timur has a stairway with n steps. The i-th step is a_i meters higher than its predecessor. The first step is a_1 meters higher than the ground, and the ground starts at 0 meters.



The stairs for the first test case.

Timur has q questions, each denoted by an integer $k_1, \ldots, k_q$. For each question k_i, you have to print the maximum possible height Timur can achieve by climbing the steps if his legs are of length k_i. Timur can only climb the j-th step if his legs are of length at least a_j. In other words, $k_i \geq a_j$ for each step j climbed.

Note that you should answer each question independently.

Input

The first line contains a single integer t ($1 \leq t \leq 100$) — the number of test cases.

The first line of each test case contains two integers n, q ($1 \leq n, q \leq 2 \cdot 10^5$) — the number of steps and the number of questions, respectively.

The second line of each test case contains n integers ($1 \leq a_i \leq 10^9$) — the height of the steps.

The third line of each test case contains q integers ($0 \leq k_i \leq 10^9$) — the numbers for each question.

It is guaranteed that the sum of n does not exceed $2 \cdot 10^5$, and the sum of q does not exceed $2 \cdot 10^5$.

Output

For each test case, output a single line containing q integers, the answer for each question.

Please note, that the answer for some questions won't fit into 32-bit integer type, so you should use at least 64-bit integer type in your programming language (like `long long` for C++).

Examples

Input	Output

```
3
4 5
1 2 1 5
1 2 4 9 10
2 2
1 1
0 1
3 1
1000000000 1000000000 1000000000
1000000000
```
	```
	1 4 4 9 9
	0 2
	3000000000
	```

Note

Consider the first test case, pictured in the statement.

- If Timur's legs have length 1, then he can only climb stair 1, so the highest he can reach is 1 meter.
- If Timur's legs have length 2 or 4, then he can only climb stairs 1, 2, and 3, so the highest he can reach is $1+2+1=4$ meters.
- If Timur's legs have length 9 or 10, then he can climb the whole staircase, so the highest he can reach is $1+2+1+5=9$ meters.

In the first question of the second test case, Timur has no legs, so he cannot go up even a single step. `:(`

Problem E. Wooden Toy Festival

Time limit 3000 ms
Mem limit 262144 kB
Problem Link

In a small town, there is a workshop specializing in woodwork. Since the town is small, only **three** carvers work there.

Soon, a wooden toy festival is planned in the town. The workshop employees want to prepare for it.

They know that n people will come to the workshop with a request to make a wooden toy. People are different and may want different toys. For simplicity, let's denote the pattern of the toy that the i-th person wants as a_i ($1 \le a_i \le 10^9$).

Each of the carvers can choose an integer pattern x ($1 \le x \le 10^9$) in advance, **different carvers can choose different patterns**. x is the integer. During the preparation for the festival, the carvers will perfectly work out the technique of making the toy of the chosen pattern, which will allow them to cut it out of wood instantly. To make a toy of pattern y for a carver who has chosen pattern x, it will take $|x - y|$ time, because the more the toy resembles the one he can make instantly, the faster the carver will cope with the work.

On the day of the festival, when the next person comes to the workshop with a request to make a wooden toy, the carvers can choose who will take on the job. At the same time, the carvers are very skilled people and can work on orders for different people **simultaneously**.

Since people don't like to wait, the carvers want to choose patterns for preparation in such a way that the **maximum** waiting time over all people is as **small** as possible.

Output the *best* maximum waiting time that the carvers can achieve.

Input

The first line of the input contains an integer t ($1 \le t \le 10^4$) — the number of test cases.

Then follow the descriptions of the test cases.

The first line of a test case contains a single integer n ($1 \le n \le 2 \cdot 10^5$) — the number of people who will come to the workshop.

The second line of a test case contains n integers $a_1, a_2, a_3, \ldots, a_n$ ($1 \le a_i \le 10^9$) — the patterns of toys.

The sum of all n values over all test cases does not exceed $2 \cdot 10^5$.

Output

Output t numbers, each of which is the answer to the corresponding test case — the *best* maximum waiting time that the carvers can achieve.

Examples

Input

```
5
6
1 7 7 9 9 9
6
5 4 2 1 30 60
9
14 19 37 59 1 4 4 98 73
1
2
6
3 10 1 17 15 11
```

Output

```
0
2
13
0
1
```

Note

In the first example, the carvers can choose patterns $1, 7, 9$ for preparation.

In the second example, the carvers can choose patterns $3, 30, 60$ for preparation.

In the third example, the carvers can choose patterns $14, 50, 85$ for preparation.

Problem F. Number Game

Time limit 2000 ms
Mem limit 262144 kB
Problem Link

Alice and Bob are playing a game. They have an array of positive integers a of size n.

Before starting the game, Alice chooses an integer $k \geq 0$. The game lasts for k stages, the stages are numbered from 1 to k. During the i-th stage, Alice must remove an element from the array that is less than or equal to $k - i + 1$. After that, if the array is not empty, Bob must add $k - i + 1$ to an arbitrary element of the array. Note that both Alice's move and Bob's move are two parts of the same stage of the game. If Alice can't delete an element during some stage, she loses. If the k-th stage ends and Alice hasn't lost yet, she wins.

Your task is to determine the maximum value of k such that Alice can win if both players play optimally. Bob plays against Alice, so he tries to make her lose the game, if it's possible.

Input

The first line contains a single integer t ($1 \leq t \leq 100$) — the number of test cases.

The first line of each test case contains a single integer n ($1 \leq n \leq 100$) — the size of the array a.

The second line contains n integers $a_1, a_2, \dots, a_n$ ($1 \leq a_i \leq n$).

Output

For each test case, print one integer — the maximum value of k such that Alice can win if both players play optimally.

Examples

Input	Output
4	2
3	0
1 1 2	1
4	3
4 4 4 4	
1	
1	
5	
1 3 2 1 1	

Problem G. K-th Not Divisible by n

Time limit 1000 ms
Mem limit 262144 kB
Problem Link

You are given two positive integers n and k. Print the k-th positive integer that is not divisible by n.

For example, if $n = 3$, and $k = 7$, then all numbers that are not divisible by 3 are: $1, 2, 4, 5, 7, 8, 10, 11, 13 \ldots$. The 7-th number among them is 10.

Input

The first line contains an integer t $(1 \leq t \leq 1000)$ — the number of test cases in the input. Next, t test cases are given, one per line.

Each test case is two positive integers n $(2 \leq n \leq 10^9)$ and k $(1 \leq k \leq 10^9)$.

Output

For each test case print the k-th positive integer that is not divisible by n.

Examples

Input	Output

```
6
3 7
4 12
2 1000000000
7 97
1000000000 1000000000
2 1
```
	```
	10
	15
	1999999999
	113
	1000000001
	1
	```

Problem H. Sum of Cubes

Time limit 2000 ms
Mem limit 262144 kB
Problem Link

You are given a positive integer x. Check whether the number x is representable as the sum of the cubes of two positive integers.

Formally, you need to check if there are two integers a and b ($1 \le a, b$) such that $a^3 + b^3 = x$.

For example, if $x = 35$, then the numbers $a = 2$ and $b = 3$ are suitable ($2^3 + 3^3 = 8 + 27 = 35$). If $x = 4$, then no pair of numbers a and b is suitable.

Input

The first line contains one integer t ($1 \le t \le 100$) — the number of test cases. Then t test cases follow.

Each test case contains one integer x ($1 \le x \le 10^{12}$).

Please note, that the input for some test cases won't fit into 32-bit integer type, so you should use at least 64-bit integer type in your programming language.

Output

For each test case, output on a separate line:

- "`YES`" if x is representable as the sum of the cubes of two positive integers.
- "`NO`" otherwise.

You can output "`YES`" and "`NO`" in any case (for example, the strings `yEs`, `yes`, `Yes` and `YES` will be recognized as positive).

Examples

Input	Output

```
7
1
2
4
34
35
16
703657519796
```
	```
	NO
	YES
	NO
	NO
	YES
	YES
	YES
	```

Note

The number 1 is not representable as the sum of two cubes.

The number 2 is represented as $1^3 + 1^3$.

The number 4 is not representable as the sum of two cubes.

The number 34 is not representable as the sum of two cubes.

The number 35 is represented as $2^3 + 3^3$.

The number 16 is represented as $2^3 + 2^3$.

The number 703657519796 is represented as $5779^3 + 7993^3$.

Problem I. Maximum Median

Time limit 2000 ms
Mem limit 262144 kB
Problem Link

You are given an array a of n integers, where n is odd. You can make the following operation with it:

- Choose one of the elements of the array (for example a_i) and increase it by 1 (that is, replace it with $a_i + 1$).

You want to make the median of the array the largest possible using at most k operations.

The median of the odd-sized array is the middle element after the array is sorted in non-decreasing order. For example, the median of the array $[1, 5, 2, 3, 5]$ is 3.

Input

The first line contains two integers n and k ($1 \le n \le 2 \cdot 10^5$, n is odd, $1 \le k \le 10^9$) — the number of elements in the array and the largest number of operations you can make.

The second line contains n integers $a_1, a_2, \ldots, a_n$ ($1 \le a_i \le 10^9$).

Output

Print a single integer — the maximum possible median after the operations.

Examples

Input	Output
3 2 1 3 5	5

Input	Output
5 5 1 2 1 1 1	3

Input	Output
7 7 4 1 2 4 3 4 4	5

Note

In the first example, you can increase the second element twice. Than array will be $[1, 5, 5]$ and it's median is 5.

In the second example, it is optimal to increase the second number and than increase third and fifth. This way the answer is 3.

In the third example, you can make four operations: increase first, fourth, sixth, seventh element. This way the array will be $[5, 1, 2, 5, 3, 5, 5]$ and the median will be 5.

Problem J. Two Cakes

Time limit 1000 ms
Mem limit 262144 kB
Problem Link

It's New Year's Eve soon, so Ivan decided it's high time he started setting the table. Ivan has bought two cakes and cut them into pieces: the first cake has been cut into a pieces, and the second one — into b pieces.

Ivan knows that there will be n people at the celebration (including himself), so Ivan has set n plates for the cakes. Now he is thinking about how to distribute the cakes between the plates. Ivan wants to do it in such a way that all following conditions are met:

1. Each piece of each cake is put on some plate;
2. Each plate contains at least one piece of cake;
3. No plate contains pieces of both cakes.

To make his guests happy, Ivan wants to distribute the cakes in such a way that the minimum number of pieces on the plate is maximized. Formally, Ivan wants to know the maximum possible number x such that he can distribute the cakes according to the aforementioned conditions, and each plate will contain at least x pieces of cake.

Help Ivan to calculate this number x!

Input

The first line contains three integers n, a and b ($1 \le a, b \le 100$, $2 \le n \le a + b$) — the number of plates, the number of pieces of the first cake, and the number of pieces of the second cake, respectively.

Output

Print the maximum possible number x such that Ivan can distribute the cake in such a way that each plate will contain at least x pieces of cake.

Input	Output
`5 2 3`	`1`

Input	Output
`4 7 10`	`3`

Note

In the first example there is only one way to distribute cakes to plates, all of them will have 1 cake on it.

In the second example you can have two plates with 3 and 4 pieces of the first cake and two plates both with 5 pieces of the second cake. Minimal number of pieces is 3.

Problem K. Factory Machines

Time limit	1000 ms
Mem limit	524288 kB

Problem Link

A factory has n machines which can be used to make products. Your goal is to make a total of t products.

For each machine, you know the number of seconds it needs to make a single product. The machines can work simultaneously, and you can freely decide their schedule.

What is the shortest time needed to make t products?

Input

The first input line has two integers n and t: the number of machines and products.

The next line has n integers $k_1, k_2, \ldots, k_n$: the time needed to make a product using each machine.

Output

Print one integer: the minimum time needed to make t products.

Constraints

- $1 \leq n \leq 2 \cdot 10^5$
- $1 \leq t \leq 10^9$
- $1 \leq k_i \leq 10^9$

Example

Input	Output
3 7 3 2 5	8

Explanation: Machine 1 makes two products, machine 2 makes four products and machine 3 makes one product.