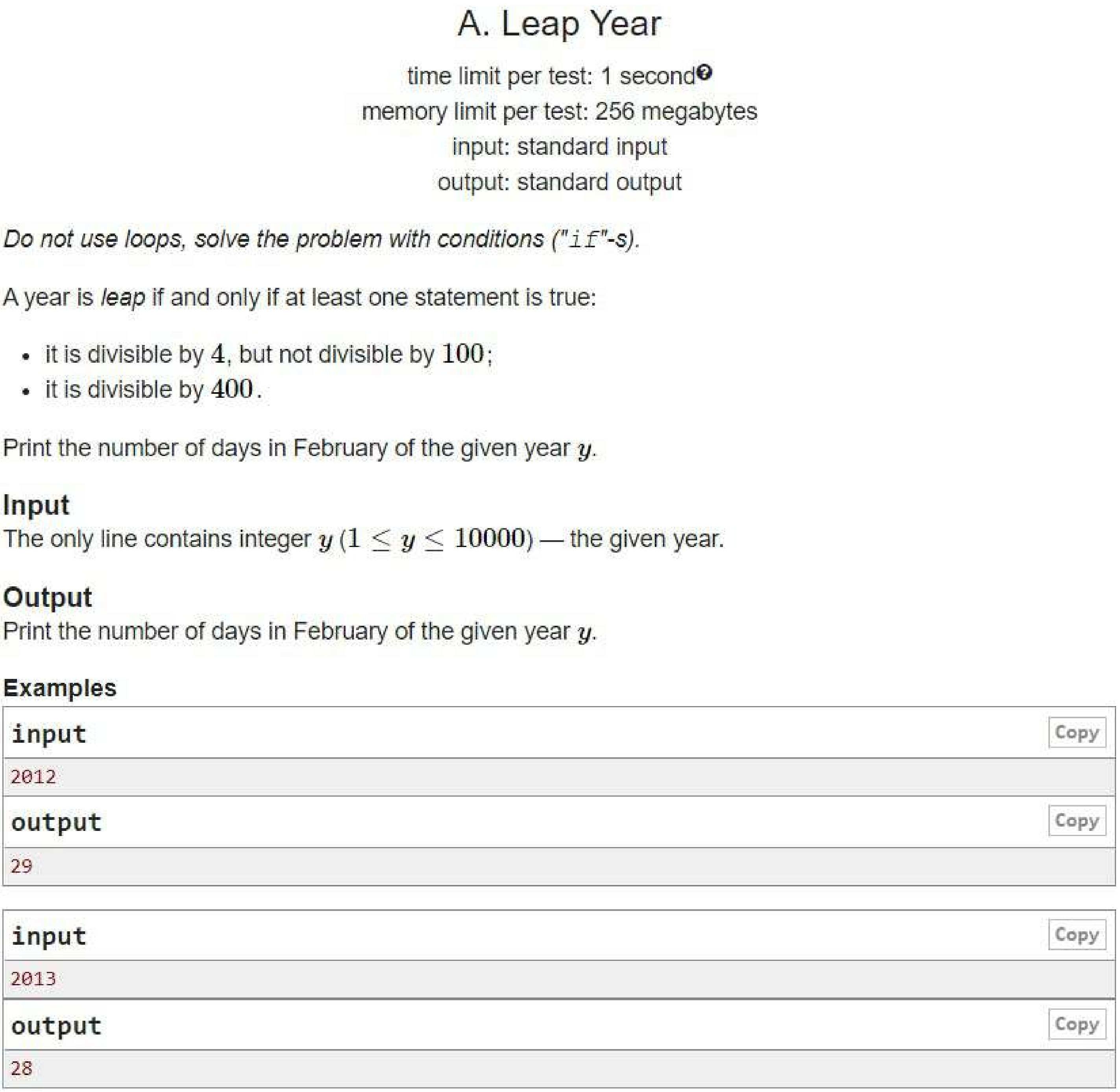


A. Leap Year

time limit per test: 1 second
memory limit per test: 256 megabytes
input: standard input
output: standard output

Do not use loops, solve the problem with conditions ("`if`"-s).

A year is *leap* if and only if at least one statement is true:

- it is divisible by 4, but not divisible by 100;
- it is divisible by 400.

Print the number of days in February of the given year y.

Input

The only line contains integer y $(1 \leq y \leq 10000)$ — the given year.

Output

Print the number of days in February of the given year y.

Examples

input

```
2012
```

output

```
29
```

input

```
2013
```

output

```
28
```

B. USB Flash Drives

time limit per test: 2 seconds

memory limit per test: 256 megabytes

input: standard input

output: standard output

Sean is trying to save a large file to a USB flash drive. He has n USB flash drives with capacities equal to $a_1, a_2, ..., a_n$ megabytes. The file size is equal to m megabytes.

Find the minimum number of USB flash drives needed to write Sean's file, if he can split the file between drives.

Input

The first line contains positive integer n $(1 \le n \le 100)$ — the number of USB flash drives.

The second line contains positive integer m $(1 \le m \le 10^5)$ — the size of Sean's file.

Each of the next n lines contains positive integer a_i $(1 \le a_i \le 1000)$ — the sizes of USB flash drives in megabytes.

It is guaranteed that the answer exists, i. e. the sum of all a_i is not less than m.

Output

Print the minimum number of USB flash drives to write Sean's file, if he can split the file between drives.

Examples

input Copy

```
3
5
2
1
3
```

output Copy

```
2
```

input Copy

```
3
6
2
3
2
```

output Copy

```
3
```

C. Decode Cyclic Shifts

time limit per test: 2 s.
memory limit per test: 256 MB
input: standard input
output: standard output

An array a of even length was encoded with the following steps:

- split it on two parts of equal lengths (the left part and the right part),
- each part is shifted cyclically to the right on p,
- after that the resulting array is shifted cyclically to the right on q.

Example of a cyclic shift: if $[4, 5, 3, 6, 1]$ is shifted cyclically to the right on 3, the resulting array is $[3, 6, 1, 4, 5]$.

You are given **encoded** array, you task is to **decode it** (i.e. to restore the original array such that if you apply the given operations on the answer array, you obtain the given one).

Input

The first line contains three integers n, p, q $((1 \le n \le 2 \cdot 10^5, 1 \le p \le \frac{n}{2}, 1 \le q \le n)$ — number of elements in the array and values of shifts. It is guaranteed, that n is even number.

The following line contains n integers $a_0, a_1, \ldots, a_{n-1}$ $(1 \le a_i \le 10^9)$ — elements of the encoded array.

Output

Print the original array before encoding as n integers in a single line.

Examples

input

Copy

```
6 2 4
1 5 6 4 2 3
```

output

Copy

```
1 2 3 4 5 6
```

input

Copy

```
2 1 1
2 1
```

output

Copy

```
1 2
```

input

Copy

```
10 3 4
20 71 58 14 86 44 66 42 11 11
```

output

Copy

```
42 11 86 44 66 58 14 11 20 71
```

D. The Best Gift

time limit per test: 2 seconds

memory limit per test: 256 megabytes

input: standard input

output: standard output

Emily's birthday is next week and Jack has decided to buy a present for her. He knows she loves books so he goes to the local bookshop, where there are n books on sale from one of m genres.

In the bookshop, Jack decides to buy *two books of different genres*.

Based on the genre of books on sale in the shop, find the number of options available to Jack for choosing two books of different genres for Emily. Options are considered different if they differ in at least one book.

The books are given by indices of their genres. The genres are numbered from 1 to m.

Input

The first line contains two positive integers n and m $(2 \le n \le 2 \cdot 10^5, 2 \le m \le 10)$ — the number of books in the bookstore and the number of genres.

The second line contains a sequence $a_1, a_2, ..., a_n$, where a_i $(1 \le a_i \le m)$ equals the genre of the i-th book.

It is guaranteed that for each genre there is at least one book of that genre.

Output

Print the only integer — the number of ways in which Jack can choose books.

It is guaranteed that the answer doesn't exceed the value $2 \cdot 10^9$.

Examples

input Copy

```
4 3
2 1 3 1
```

output Copy

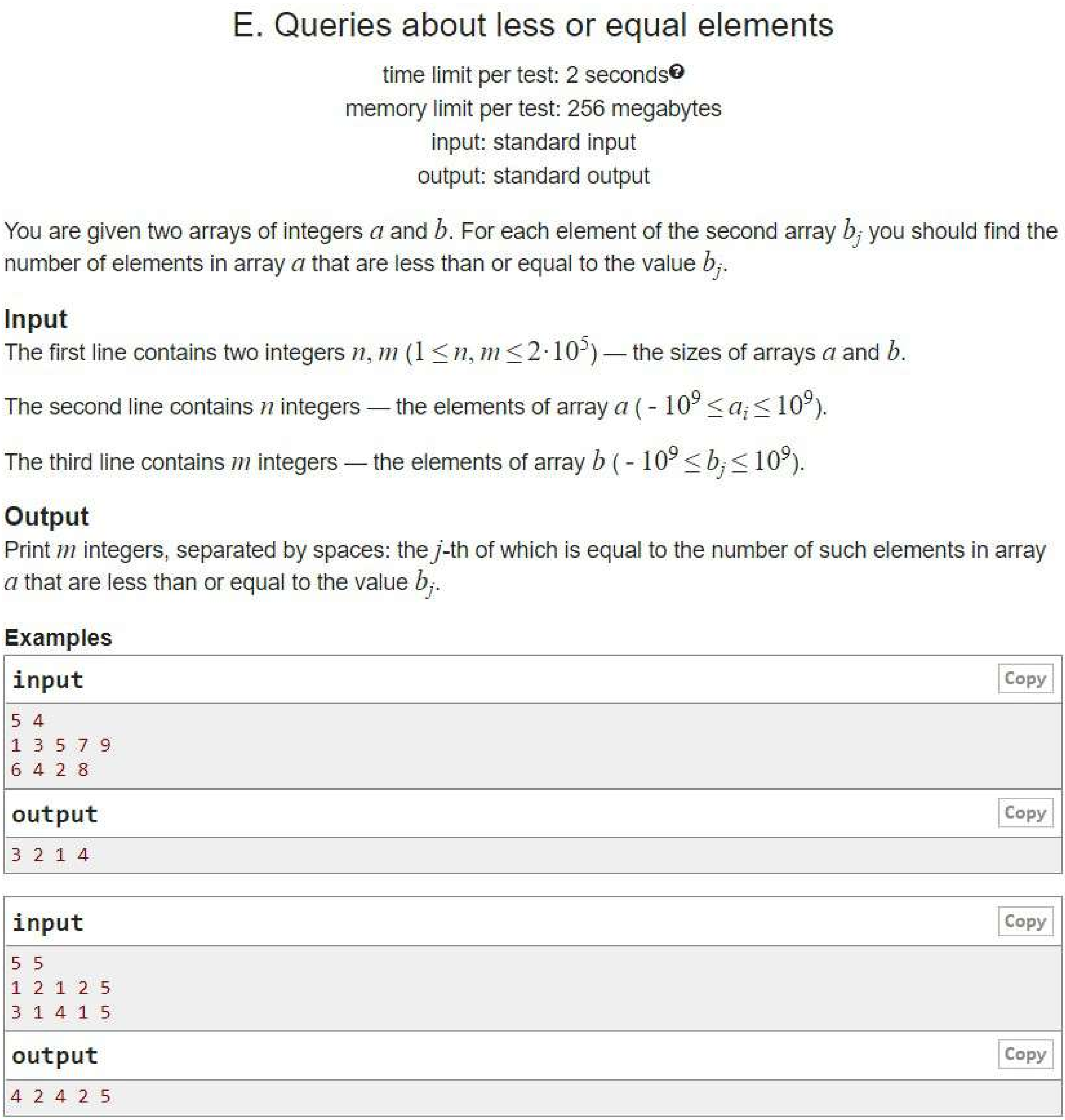
```
5
```

input Copy

```
7 4
4 2 3 1 2 4 3
```

output Copy

```
18
```



E. Queries about less or equal elements

time limit per test: 2 seconds

memory limit per test: 256 megabytes

input: standard input

output: standard output

You are given two arrays of integers a and b. For each element of the second array b_j you should find the number of elements in array a that are less than or equal to the value b_j.

Input

The first line contains two integers n, m ($1 \le n, m \le 2 \cdot 10^5$) — the sizes of arrays a and b.

The second line contains n integers — the elements of array a ($-10^9 \le a_i \le 10^9$).

The third line contains m integers — the elements of array b ($-10^9 \le b_j \le 10^9$).

Output

Print m integers, separated by spaces: the j-th of which is equal to the number of such elements in array a that are less than or equal to the value b_j.

Examples

input

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

output

```
3 2 1 4
```

input

```
5 5
1 2 1 2 5
3 1 4 1 5
```

output

```
4 2 4 2 5
```

F. Load Balancing

time limit per test: 2 seconds❓

memory limit per test: 256 megabytes

input: standard input

output: standard output

In the school computer room there are n servers which are responsible for processing several computing tasks. You know the number of scheduled tasks for each server: there are m_i tasks assigned to the i-th server.

In order to balance the load for each server, you want to reassign some tasks to make the difference between the most loaded server and the least loaded server as small as possible. In other words you want to minimize expression m_a - m_b, where a is the most loaded server and b is the least loaded one.

In one second you can reassign a single task. Thus in one second you can choose any pair of servers and move a single task from one server to another.

Write a program to find the minimum number of seconds needed to balance the load of servers.

Input

The first line contains positive number n $(1 \le n \le 10^5)$ — the number of the servers.

The second line contains the sequence of non-negative integers $m_1, m_2, ..., m_n$ $(0 \le m_i \le 2 \cdot 10^4)$, where m_i is the number of tasks assigned to the i-th server.

Output

Print the minimum number of seconds required to balance the load.

Examples

input Copy

```
2
1 6
```

output Copy

```
2
```

input Copy

```
7
10 11 10 11 10 11 11
```

output Copy

```
0
```

input Copy

```
5
1 2 3 4 5
```

output Copy

```
3
```

G. Make Palindrome

time limit per test: 2 seconds

memory limit per test: 256 megabytes

input: standard input

output: standard output

A string is called palindrome if it reads the same from left to right and from right to left. For example "`kazak`", "`oo`", "`r`" and "`mikhailrubinchikkihcniburliahkim`" are palindroms, but strings "`abb`" and "`ij`" are not.

You are given string s consisting of lowercase Latin letters. At once you can choose any position in the string and change letter in that position to any other lowercase letter. So after each changing the length of the string doesn't change. At first you can change some letters in s. Then you can permute the order of letters as you want. Permutation doesn't count as changes.

You should obtain palindrome with the minimal number of changes. If there are several ways to do that you should get the lexicographically (alphabetically) smallest palindrome. So firstly you should minimize the number of changes and then minimize the palindrome lexicographically.

Input

The only line contains string s $(1 \le |s| \le 2 \cdot 10^5)$ consisting of only lowercase Latin letters.

Output

Print the lexicographically smallest palindrome that can be obtained with the minimal number of changes.

Examples

input Copy

```
aabc
```

output Copy

```
abba
```

input Copy

```
aabcd
```

output Copy

```
abcba
```

H. Gadgets for dollars and pounds

time limit per test: 2 seconds

memory limit per test: 256 megabytes

input: standard input

output: standard output

Nura wants to buy k gadgets. She has only s burles for that. She can buy each gadget for dollars or for pounds. So each gadget is selling only for some type of currency. The type of currency and the cost in that currency are not changing.

Nura can buy gadgets for n days. For each day you know the exchange rates of dollar and pound, so you know the cost of conversion burles to dollars or to pounds.

Each day (from 1 to n) Nura can buy some gadgets by current exchange rate. Each day she can buy any gadgets she wants, but each gadget can be bought no more than once during n days.

Help Nura to find the minimum day index when she will have k gadgets. Nura always pays with burles, which are converted according to the exchange rate of the purchase day. Nura can't buy dollars or pounds, she always stores only burles. Gadgets are numbered with integers from 1 to m in order of their appearing in input.

Input

First line contains four integers n, m, k, s $(1 \le n \le 2 \cdot 10^5, 1 \le k \le m \le 2 \cdot 10^5, 1 \le s \le 10^9)$ — number of days, total number and required number of gadgets, number of burles Nura has.

Second line contains n integers a_i $(1 \le a_i \le 10^6)$ — the cost of one dollar in burles on i-th day.

Third line contains n integers b_i $(1 \le b_i \le 10^6)$ — the cost of one pound in burles on i-th day.

Each of the next m lines contains two integers t_i, c_i $(1 \le t_i \le 2, 1 \le c_i \le 10^6)$ — type of the gadget and it's cost. For the gadgets of the first type cost is specified in dollars. For the gadgets of the second type cost is specified in pounds.

Output

If Nura can't buy k gadgets print the only line with the number `-1`.

Otherwise the first line should contain integer d — the minimum day index, when Nura will have k gadgets. On each of the next k lines print two integers q_i, d_i — the number of gadget and the day gadget should be bought. All values q_i should be different, but the values d_i can coincide (so Nura can buy several gadgets at one day). The days are numbered from 1 to n.

In case there are multiple possible solutions, print any of them.

Examples

input Copy

```
5 4 2 2
1 2 3 2 1
3 2 1 2 3
1 1
2 1
1 2
2 2
```

output Copy

```
3
1 1
2 3
```