A. Years

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

During one of the space missions, humans have found an evidence of previous life at one of the planets. They were lucky enough to find a book with birth and death years of each individual that had been living at this planet. What's interesting is that these years are in the range $(1, 10^9)$! Therefore, the planet was named Longlifer.

In order to learn more about Longlifer's previous population, scientists need to determine the year with maximum number of individuals that were alive, as well as the number of alive individuals in that year. Your task is to help scientists solve this problem!

Input

The first line contains an integer n ($1 \le n \le 10^5$) — the number of people.

Each of the following n lines contain two integers b and d ($1 \le b < d \le 10^9$) representing birth and death year (respectively) of each individual.

Output

Print two integer numbers separated by blank character, y — the year with a maximum number of people alive and k — the number of people alive in year y.

In the case of multiple possible solutions, print the solution with minimum year.

Examples

input

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

output

```
2 2
```

input

```
4
3 4
4 5
4 6
8 10
```

output

```
4 2
```

Note

You can assume that an individual living from b to d has been born at the beginning of b and died at the beginning of d, and therefore living for d - b years.

B. Sorted Adjacent Differences

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

You have array of n numbers $a_1, a_2, \ldots, a_n$.

Rearrange these numbers to satisfy $|a_1 - a_2| \le |a_2 - a_3| \le \ldots \le |a_{n-1} - a_n|$, where $|x|$ denotes absolute value of x. It's always possible to find such rearrangement.

Note that all numbers in a are not necessarily different. In other words, some numbers of a may be same.

You have to answer independent t test cases.

Input

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

The first line of each test case contains single integer n ($3 \le n \le 10^5$) — the length of array a. It is guaranteed that the sum of values of n over all test cases in the input does not exceed 10^5.

The second line of each test case contains n integers $a_1, a_2, \ldots, a_n$ ($-10^9 \le a_i \le 10^9$).

Output

For each test case, print the rearranged version of array a which satisfies given condition. If there are multiple valid rearrangements, print any of them.

Example

input

```
2
6
5 -2 4 8 6 5
4
8 1 4 2
```

output

```
5 5 4 6 8 -2
1 2 4 8
```

Note

In the first test case, after given rearrangement, $|a_1 - a_2| = 0 \le |a_2 - a_3| = 1 \le |a_3 - a_4| = 2 \le |a_4 - a_5| = 2 \le |a_5 - a_6| = 10$. There are other possible answers like "`5 4 5 6 -2 8`".

In the second test case, after given rearrangement, $|a_1 - a_2| = 1 \leq |a_2 - a_3| = 2 \leq |a_3 - a_4| = 4$. There are other possible answers like "`2 4 8 1`".

C. More Cowbell

time limit per test: 2 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

Kevin Sun wants to move his precious collection of n cowbells from Naperthrill to Exeter, where there is actually grass instead of corn. Before moving, he must pack his cowbells into k boxes of a fixed size. In order to keep his collection safe during transportation, he won't place more than **two** cowbells into a single box. Since Kevin wishes to minimize expenses, he is curious about the smallest size box he can use to pack his entire collection.

Kevin is a meticulous cowbell collector and knows that the size of his i-th $(1 \leq i \leq n)$ cowbell is an integer s_i. In fact, he keeps his cowbells sorted by size, so $s_{i-1} \leq s_i$ for any $i > 1$. Also an expert packer, Kevin can fit one or two cowbells into a box of size s if and only if the sum of their sizes does not exceed s. Given this information, help Kevin determine the smallest s for which it is possible to put all of his cowbells into k boxes of size s.

Input

The first line of the input contains two space-separated integers n and k $(1 \leq n \leq 2 \cdot k \leq 100\,000)$, denoting the number of cowbells and the number of boxes, respectively.

The next line contains n space-separated integers $s_1, s_2, \ldots, s_n$ $(1 \leq s_1 \leq s_2 \leq \ldots \leq s_n \leq 1\,000\,000)$, the sizes of Kevin's cowbells. It is guaranteed that the sizes s_i are given in non-decreasing order.

Output

Print a single integer, the smallest s for which it is possible for Kevin to put all of his cowbells into k boxes of size s.

Examples

input

```
2 1
2 5
```

output

```
7
```

input

```
4 3
2 3 5 9
```

output

```
9
```

input

```
3 2
3 5 7
```

output

```
8
```

Note

In the first sample, Kevin must pack his two cowbells into the same box.

In the second sample, Kevin can pack together the following sets of cowbells: $\{2, 3\}$, $\{5\}$ and $\{9\}$.

In the third sample, the optimal solution is $\{3, 5\}$ and $\{7\}$.

D. The Parade

time limit per test: 2 seconds
memory limit per test: 512 megabytes
input: standard input
output: standard output

The Berland Army is preparing for a large military parade. It is already decided that the soldiers participating in it will be divided into k rows, and all rows will contain *the same* number of soldiers.

Of course, not every arrangement of soldiers into k rows is suitable. Heights of all soldiers in the same row should not differ by more than 1. The height of each soldier is an integer between 1 and n.

For each possible height, you know the number of soldiers having this height. To conduct a parade, you have to choose the soldiers participating in it, and then arrange *all of the chosen soldiers* into k rows so that both of the following conditions are met:

- each row has the same number of soldiers,
- no row contains a pair of soldiers such that their heights differ by 2 or more.

Calculate the maximum number of soldiers who can participate in the parade.

Input

The first line contains one integer t $(1 \leq t \leq 10000)$ — the number of test cases. Then the test cases follow.

Each test case begins with a line containing two integers n and k $(1 \leq n \leq 30000,\ 1 \leq k \leq 10^{12})$ — the number of different heights of soldiers and the number of rows of soldiers in the parade, respectively.

The second (and final) line of each test case contains n integers $c_1, c_2, \ldots, c_n$ $(0 \leq c_i \leq 10^{12})$, where c_i is the number of soldiers having height i in the Berland Army.

It is guaranteed that the sum of n over all test cases does not exceed 30000.

Output

For each test case, print one integer — the maximum number of soldiers that can participate in the parade.

Example

input

```
5
3 4
7 1 13
1 1
100
1 3
100
2 1
1000000000000 1000000000000
4 1
10 2 11 1
```

output

```
16
100
99
2000000000000
13
```

Note

Explanations for the example test cases:

1. the heights of soldiers in the rows can be: $[3, 3, 3, 3]$, $[1, 2, 1, 1]$, $[1, 1, 1, 1]$, $[3, 3, 3, 3]$ (each list represents a row);
2. all soldiers can march in the same row;
3. 33 soldiers with height 1 in each of 3 rows;
4. all soldiers can march in the same row;
5. all soldiers with height 2 and 3 can march in the same row.

E. Building Bridge

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

Two villages are separated by a river that flows from the north to the south. The villagers want to build a bridge across the river to make it easier to move across the villages.

The river banks can be assumed to be vertical straight lines $x = a$ and $x = b$ $(0 < a < b)$.

The west village lies in a steppe at point $O = (0, 0)$. There are n pathways leading from the village to the river, they end at points $A_i = (a, y_i)$. The villagers there are plain and simple, so their pathways are straight segments as well.

The east village has reserved and cunning people. Their village is in the forest on the east bank of the river, but its exact position is not clear. There are m twisted paths leading from this village to the river and ending at points $B_i = (b, y'_i)$. The lengths of all these paths are known, the length of the path that leads from the eastern village to point B_i, equals l_i.

The villagers want to choose exactly one point on the left bank of river A_i, exactly one point on the right bank B_j and connect them by a straight-line bridge so as to make the total distance between the villages (the sum of $|OA_i| + |A_iB_j| + l_j$, where $|XY|$ is the Euclidean distance between points X and Y) were minimum. The Euclidean distance between points (x_1, y_1) and (x_2, y_2) equals $\sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$.

Help them and find the required pair of points.

Input

The first line contains integers n, m, a, b $(1 \le n, m \le 10^5, 0 < a < b < 10^6)$.

The second line contains n integers in the ascending order: the i-th integer determines the coordinate of point A_i and equals y_i $(|y_i| \le 10^6)$.

The third line contains m integers in the ascending order: the i-th integer determines the coordinate of point B_i and equals y'_i $(|y'_i| \le 10^6)$.

The fourth line contains m more integers: the i-th of them determines the length of the path that connects the eastern village and point B_i, and equals l_i $(1 \le l_i \le 10^6)$.

It is guaranteed, that there is such a point C with abscissa at least b, that $|B_iC| \le l_i$ for all i $(1 \le i \le m)$. It is guaranteed that no two points A_i coincide. It is guaranteed that no two points B_i coincide.

Output

Print two integers — the numbers of points on the left (west) and right (east) banks, respectively, between which you need to build a bridge. You can assume that the points on the west bank are numbered from 1 to n, in the order in which they are given in the input. Similarly, the points on the east bank are numbered from 1 to m in the order in which they are given in the input.

If there are multiple solutions, print any of them. The solution will be accepted if the final length of the path will differ from the answer of the jury by no more than 10^{-6} in absolute or relative value.

Examples

input

```
3 2 3 5
-2 -1 4
-1 2
7 3
```

output

```
2 2
```