CP1 [2023] Make-up Lab

A. Research groups

1 second, 256 megabytes

Young students from different schools are participating in a science fair organized by the renowned scientist, Dr. Smith. Each student has a research score, represented by a positive integer r_i, indicating their level of scientific expertise. Dr. Smith wants to group the students into teams for the fair.

If the research score of a researcher is r, then he can only be part of team whose size is greater than or equal r. Dr. Smith is concerned about efficiently forming the teams and seeks your help.

Input

The first line contains the number of independent test cases T ($1 \le T \le 2 \cdot 10^5$).

Next $2T$ lines contain descriptions of test cases.

For each test case, the first line contains the number of students N ($1 \le N \le 2 \cdot 10^5$).

The second line contains N integers $r_1, r_2, \ldots, r_N (1 \le r_i \le N)$, representing the research scores of the i-th student.

It's guaranteed that the sum of all N doesn't exceed $3 \cdot 10^5$.

Output

Print T numbers, each number on a separate line.

In the i-th line, print the maximum number of teams Dr. Smith can form in the i-th test case.

input

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

output

```
2
3
```

B. Alchemist

1 second, 256 megabytes

You are a skilled alchemist on a quest to transform a set of magical crystals into gold coins. You have an array of crystals represented by integers $crystals$ of size n. Initially, all elements of the $crystals$ array are equal to 1, symbolizing the basic form of the crystals.

You can perform a special transformation operation on a crystal by choosing an index i $(1 \le i \le n)$ and a positive integer x $(x > 0)$. When you apply the transformation, the value of the crystal at index i, $crystals[i]$, increases by $\lfloor crystals[i]/x \rfloor$. This means that you enhance the magical properties of the crystal.

After performing all the operations, you will receive a certain number of gems for each crystal whose final value matches a target value given by another array $target\_values$, denoted by $target\_values[i]$, and this will yield $gems[i]$ gems.

You have to maximize the total number of gems you can obtain after atmost k operations.

Input

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

For each test case:

- The first line contains two integers n and k $(1 \le n \le 103; 0 \le k \le 10^6)$ — the size of the crystal array and the maximum number of operations, respectively.

- The second line contains n integers $target\_values[1], target\_values[2], \ldots, target\_values[n]$ $(1 \le target\_values[i] \le 103)$ representing the desired final values of the crystals.

- The third line contains n integers $gems[1], gems[2], \ldots, gems[n]$ $(1 \le gems[i] \le 10^6)$ representing the number of gems you receive for each corresponding crystal whose final value matches the target value.

The sum of n over all test cases does not exceed 10^3.

Output

Print the number of gems you can obtain for each testcase in separate lines.

input

```
2
2 0
3 2
5 7
5 5
1 9 3 1 2
3 5 3 1 1
```

output

```
0
10
```

C. Building Bridges

1.5 seconds, 1024 megabytes

You are playing a game where you, the general of your army, need to prepare to defend your country, Zephyria, from its sworn enemy, Valeria.

Your country is composed of a group of n islands, and to improve the logistics of your army, you need to ensure that every island is connected to every other island through a sequence of bridges.

You can build a bridge from island x to y at cost of c gold coins as long as you have upgraded your Engineering barracks to level l.

You start the game with Engineering barracks at level 0. You are given an array a of length k where a_i represents the cost in runes to upgrade barracks from level $i-1$ to i and k is the highest level of barracks available.

You can currently have r units of runes and g units of gold. Minimise the amount of runes you need to connect all the islands of your nation.

Input

The first line contains 5 integers, n, the number of islands, m, the number of possible bridges your Engineering corps can build, $k (1 \le n, m, k \le 10^5)$, the maximum level of Engineering barracks, r, the units of runes and $g (1 \le r, g \le 10^9)$, the units of gold available in your treasury.

The next line contains the array $a(1 \le a_i \le 10^9)$ of length k, denoting the cost in runes to upgrade the barracks.

The next m lines contain 4 integers each, x, $y(1 \le x, y \le n)$, $c(1 \le c \le 10^9)$ and $l(1 \le l \le k)$.

Output

Find the minimum number of runes you need to spend to connect all the islands in your country. If impossible, output -1.

input
```
5 4 10 1000 1000
1 1 1 1 1 1 1 1 1 1
1 2 100 5
3 2 100 9
2 4 100 3
1 5 100 7
```
output
```
9
```

input
```
3 8 3 1000 10
1 1 1
1 2 100 1
1 3 100 1
1 2 50 2
2 3 100 2
3 1 200 2
1 2 1 3
1 3 1 3
2 3 1 3
```
output
```
3
```

D. Clever Pairs in Sequence.

1 second, 256 megabytes

The problem involves an array $'p'$, denoted as $p_1, p_2, \ldots, p_n$. Let's define $f(l, r, x)$ as the count of indices $'k'$ satisfying the condition: $l \le k \le r$ and p_k = x. Your task is to determine the number of pairs of indices (i, j) with the condition that $1 \le i < j \le n$, and $f(1, i, p_i) > f(j, n, p_j)$.

In simpler terms, you need to find the number of pairs of positions (i, j) in the sequence, where the count of occurrences of a_i in the prefix of the sequence (from the beginning to position i) is greater than the count of occurrences of a_j in the suffix of the sequence (from position j to the end).

Input

The first line of the input contains an integer n ($1 \le n \le 10^6$). The second line contains n space-separated integers $a_1, a_2, \ldots, a_n$ ($0 \le a_i \le 10^9$).

Output

Print a single integer — the answer to the problem.

input
```
3
1 1 1
```
output
```
1
```

E. Eccentric Wizard

1 second, 256 megabytes

You are travelling to lands of Celestoria to meet a wizard and purchase k ancient artifacts you desperately need. On your way there you come across another returning traveler who tells you how he got robbed by the wizard as he kept changing prices of remaining artifacts everytime he made an purchase.

Upon further inquiry, you learn that the wizard adds a markup or discount on artifacts based on the previous artifact purchased. You manage to find the base price of each of the n artifacts the wizard has and also obtain a $n \times n$ matrix $disc$ where $disc_{x,y}$ indicates discount on x^{th} artifact if you had previously purchased artifact y.

A negative value of $disc_{x,y}$ indicates a markup. If the price p of artifact is negative after applying the discount, the wizard will pay you $abs(p)$ for buying the artifact.

Since these are very rare artifacts, even the wizard only possesses single specimen of every artifact. Can you find the minimum price required to obtain the k artifacts you need ?

Input

The first line of each testcase contains 2 integers, $n(1 \le n \le 14)$, the number of artifacts the wizard has, and $k(1 \le k \le n)$, the number of artifacts you want.

The next line contains an array $a(1 \le a_i \le n\, \forall i \in [1, k])$ of length k, a_i indicating that you need the i^{th} artifact.

The next line contains an array p of n integers, p_i denoting the base price of i^{th} artifact $(1 \le p_i \le 10^9)$.

The next n lines contain the matrix $disc$ with n integers in each line. The x^{th} row in the matrix has n integers denoting the discount available on artifact x for buying each of the n artifacts immediately before $(-10^9 \le disc_{x,y} \le 10^9)$.

Output

Print the minimum price you need to pay to buy the all the k artifacts. Incase you are able to earn money, output the profit as negative value.

input
```
3 2
1 2
1 2 3
0 0 0
0 0 0
0 0 0
```
output
```
3
```

input
```
3 2
1 2
1 2 3
0 -1 0
0 0 10
2 0 0
```
output
```
-6
```

input

```
5 2
4 5
1 10000 100 10 10
0 0 0 0 0
0 0 0 0 0
1000 0 0 0 0
0 0 0 0 0
0 0 0 0 0
```

output

```
-879
```

In the first case, the order in which you purchase the artifacts does not make any difference.

In the second case, if we purchase artifact 1 and then 2, it costs 3, and purchasing artifact 2 and then 1 costs 4 units due to markup of 1. But if we buy the artifact 3, which we didn't even need, in order $1 \to 3 \to 2$, then we can make a profit of 6.

In the third test, purchasing artifacts 1 and 3 (even though they are not needed) leads to profit of 899, after which we can purchase 4 and 5 artifacts afterwards.