

Problem A. Group of Permutations

Input file: `standard input`
Output file: `standard output`
Time limit: 1 second
Memory limit: 256 megabytes

You are given an array a of n positive integers.

Can you partition this array into consecutive groups such that every group is a permutation of any size ?

A permutation is an array consisting of n distinct integers from 1 to n in arbitrary order. For example, $[3, 1, 2, 5, 4]$ is a permutation, but $[1, 2, 1]$ is not a permutation (1 appears twice in the array) and $[2, 1, 4]$ is also not a permutation ($n = 3$, but 4 is in the array).

For example, if $a = [4, 1, 3, 2, 2, 1, 3, 2, 1]$ you can split it into 3 groups : $([4, 1, 3, 2] [2, 1] [3, 2, 1])$. However the array $a = [4, 1, 3, 2, 3, 1]$ can't be partitioned into groups of permutations.

Input

The first line of each test case contains one integer n ($1 \leq n \leq 2000$) — the size of a

The second line contains n positive integers $a_1, a_2, \dots, a_n$ ($1 \leq a_i \leq 2000$)

Output

If there is a solution, print "YES". Otherwise, print "NO"

Examples

standard input	standard output
9 4 1 3 2 2 1 3 2 1	YES
6 4 1 3 2 3 1	NO

Problem B. Simulation

Input file: **standard input**
Output file: **standard output**
Time limit: 1.5 seconds
Memory limit: 64 megabytes

Rami and Oussama frequently talk about simulations, and how on many occasions, order arises from disorder during many simulations.

To test this observation, Oussama conceived the following simulation:

Initially, he will have an array A of size n **filled with zeros**. Then, he will do m times these two steps:

1. Select a number k uniformly in $\{1, \dots, n\}$
2. Increment A_k by 1

Upon investigating the simulation, he noted that sometimes, there is at least one element which grows too much. So he wanted to know how likely such an event occurs by doing the following:

1. He chose the size of the array n , and a threshold K .
2. He applied m steps of the simulation on this array, and memorized the content of the array at the final state.
3. Finally, He masked the content of the array, and invited Rami to guess the probability that **there exists at least** one element of A greater or equal to the chosen threshold K .

As the question is a little bit hard, Oussama asked you to help Rami about it.

Input

One line containing 3 integers: n, m, K where:

- $1 \leq n \leq 300$: the size of the array A .
- $0 \leq m \leq 300$: the number of steps of the simulation.
- $0 \leq K \leq m$: the threshold.

Output

print a real number $0 \leq p \leq 1$ the probability that at least one of the elements of A is greater than or equal to K after m steps of the simulation.

Examples

standard input	standard output
2 10 6	0.753906
300 300 5	0.671265
10 3 2	0.28

Note

For the first test case, we have a simulation with an array of size 2, and 10 iterations. The threshold is $K \geq 6$, the only case when $\max(A_1, A_2) < 6$ is when $A_1 = A_2 = 5$. There are $\binom{10}{5}$ possible scenarios where $A_1 = A_2 = 5$ out of the 2^{10} possible scenarios. Thus the probability that $A_1 \geq 6$ or $A_2 \geq 6$ is :

$$p = 1 - \frac{1}{2^{10}} \times \binom{10}{5} = 1 - \frac{10!}{(5!)^2 2^{10}} = \frac{193}{256} \approx 0.75390625$$

Where $\binom{a}{b} = \frac{a!}{b!(a-b)!}$ is the binomial coefficient.

Problem C. Co-sortable Strings

Input file: **standard input**
Output file: **standard output**
Time limit: 1 second
Memory limit: 128 megabytes

Rami, Yessine and Oussama recently participated on a competitive programming contest. And after finishing it, they were tired. So, to have some fun, they decided to go to the café at 16:00.

Usually, Oussama is the first one to come, and he will always wait for Rami and Yessine to come 15 to 30 minutes late. But this time, Rami and Yessine are at the café and Oussama did not come yet.

As this situation is a little bit strange, Yessine called Oussama to check where he is, and of course, Oussama told him that he is ready, but he is just checking why his greedy solution was not accepted in the contest, and once he knows what is wrong with it, he will come.

Now, Yessine knows very well that this may take some time, so he invented a little challenge with Rami that will kill their time:

1. First of all, he created two strings A and B both of length L and having only **lowercase** characters:
2. Then, for two substring $A[l \dots r]$, $B[l \dots r]$ of A and B respectively, he called them **co-sortable** if they can be simultaneously sorted using the following steps:
 - Choose $i \in \{l, \dots, r\}$ and swap $A[i]$ and $B[i]$.
 - Repeat the rule above as many times as you want.
3. Now he asked Rami Q questions of the same type. That is , the i^{th} question takes two parameters $l_i \leq r_i$ and is of the form: **Are the substrings $A[l_i \dots r_i]$ and $B[l_i \dots r_i]$ co-sortable?**

Rami was overwhelmed by the number of such questions, so he asked you to help him answer the question so that Yessine won't win the challenge.

Input

1. The first line contains two integers L, Q with $1 \leq L \leq 10^5$ the length of the two strings, and $1 \leq Q \leq 10^5$ the number of questions
2. The next two lines contain the **lowercase alphabetic** strings A and B respectively.
3. The next Q lines contains each two integers, with the i^{th} line having the parameters $1 \leq l_i \leq r_i \leq L$

Output

Q lines, with the i^{th} line equal to **YES** if the two substrings $A[l_i \dots r_i]$ and $B[l_i \dots r_i]$ are **co-sortable**. Else output **NO**.

Example

standard input	standard output
3 3	YES
cbc	YES
adc	NO
1 2	
1 1	
1 3	

Note

For the first question, $A[1, 2] = cb$ and $B[1, 2] = 'ad'$. Initially they are not sorted, but by swapping $A[1]$ and $B[1]$ we will have $'ab'$ and $'cd'$ which are both sorted. So $A[1, 2]$ and $B[1, 2]$ are co-sortable.

Problem D. Decrypting the Password

Input file: **standard input**
Output file: **standard output**
Time limit: 1.2 seconds
Memory limit: 256 megabytes

Yessine was trying to log into his email when he realized that he forgot his password. Fortunately, he wrote it in a text file and saved it just in case. However, when he opened the file, he didn't find the password but rather a very long string of digits. He then remembered that he didn't write the password but rather an encryption of it. As a matter of fact, the password is the number of substrings whose numerical representation is divisible by 11. Yessine wants to open his email, help him decrypt the password.

A string a is a substring of a string b if a can be obtained from b by deletion of several (possibly, zero or all) characters from the beginning and several (possibly, zero or all) characters from the end.

Input

The first line contains one integer t ($1 \leq t \leq 10^6$) — the number of test cases.

The first line of each test case contains one integer n ($1 \leq n \leq 10^6$) — the length of the string s .

The second line of each test case contains a string consisting of n decimal digits.

It is guaranteed that the sum of n over all test cases does not exceed 10^6 .

Output

Print one line containing the answer: The number of substrings whose decimal representation is divisible by 11

Example

standard input	standard output
4	1
3	4
121	0
4	3
1111	
6	
123456	
8	
20654301	

Note

- In the first case, the only substring divisible by 11 is 121
- In the second case, the string 1111 contains 3 substrings whose decimal representation is 11. Also $1111 = 11 \times 101$ is divisible by 11. Finally we can verify that 111 is not divisible by 11. Therefore, the total number of such substrings is 4.

Problem E. Exam

Input file: standard input
Output file: standard output
Time limit: 2 seconds
Memory limit: 256 megabytes

Yessine was in the middle of an exam when he got bored. He doesn't really like exams or studying, but he likes problem-solving so he unconsciously started thinking about random problems. At that moment, one particular problem came to his mind:

You have to make an $n \times n$ grid consisting of white cells and black cells. There are only two constraints:

1. The number of black cells on the i^{th} column is C_i
2. The number of black cells on the i^{th} row is R_i

Yessine would've loved to solve this problem, but he remembered he was taking an exam and he must answer the questions or he would fail, so he left it to you.

Input

The first line contains one integer t ($1 \leq t \leq 10^5$) — the number of test cases.

The first line of each test case contains the integer $1 \leq n \leq 2000$

The second line of each test case contains n space separated integers $0 \leq C_1, \dots, C_n \leq n$

The third line of each test case contains n space separated integers $0 \leq R_1, \dots, R_n \leq n$

It is guaranteed that the sum of n^2 over all test cases does not exceed $4 \cdot 10^6$

Output

If there exists such $n \times n$ grid, output **any** (black cells are represented by 1 and white cells are represented by 0) else output -1

Example

standard input	standard output
3	0 1 1 1
4	1 1 1 1
3 3 3 2	1 0 0 0
3 4 1 3	1 1 1 0
3	-1
3 3 1	1 1 0 0
3 3 2	0 0 1 1
4	0 1 1 1
2 2 3 2	1 0 1 0
2 2 3 2	

Note

For the second test case, let's suppose that such 3×3 matrix A exists. As $R_1 = R_2 = C_1 = C_2 = 3$,
necessarily, we have $A = \begin{matrix} & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & * & \end{matrix}$. This implies that $C_3 \geq 2$, which is a contradiction as $C_3 = 1$.

Therefore, such matrix does not exist.

Problem F. 800

Input file: **standard input**
Output file: **standard output**
Time limit: 1 second
Memory limit: 256 megabytes

Oussama, Rami and Yessine were travelling to Egypt in order to participate in a programming contest. Unfortunately, the plane crashed in some unknown island and everyone died except the three of them. They spent the next few months eating coconuts and sleeping on the sands. One day, Yessine found a bottle on the beach. Inside the bottle there was a letter. They found the number 800 written on the back of paper so they were curious about it so they started reading the letter. They were shocked to find out that the letter contained some hard problem. Oussama got angry and decided to burn the paper but Rami was really curious about the number 800, and he decided to solve it hoping that he might find an explanation. The problem says: You are given a grid of lowercase characters 'a'-'z'. You can do the following operation any number of times :

- choose a cell (i, j) and a positive integer k , and change the content of the cell $(i - k, j - k)$ or $(i - k, j + k)$ to that of (i, j) .

Given a character C , can you make the grid composed only of C ?

Rami's stomach hurts from eating too much coconuts so he can't think properly, help him solve the problem.

Input

- The first line of the input contains $1 \leq N \leq 10^3$ and $1 \leq M \leq 10^3$: the dimensions of the grid
- The Second line contains the character C
- Next N lines contain each M characters. the j th character on the i th line denotes the character at $\text{cell}(i, j)$.

Output

print a single line containing YES if you can or NO if you can't.

Examples

standard input	standard output
3 3 b aaa aaa aaa	NO
1 1 a a	YES

Note

Note that the top left cell of the grid is $(0, 0)$. And the cell $(N - 1, M - 1)$ is the one on the bottom right..

Problem G. Game of Cards

Input file: `standard input`
Output file: `standard output`
Time limit: 1.5 seconds
Memory limit: 512 megabytes

Rami and Yessine were learning some advanced data structures. Eventually, they got tired from writing code so they decided to take a break and play a game.

Rami told Yessine about a game of cards he has invented. In this game, each player will get N cards. Each card has a number written on it and both players can see each other's cards.

They will take turns to play. At each turn, a player must choose a prime number P , then, the other player must find a card in his hands with a number C divisible by P , then change the number written on that paper to $\frac{C}{P}$. The chosen prime number P should be **less than or equal to** L , where L is a number they agree on before the game starts. The player who can't make a move loses.

Rami will start first. Assuming both players will play optimally, determine who will win at each game .

Input

- The first line contains 2 integers $1 \leq N \leq 10^5$ and $2 \leq L \leq 10^6$.
- The second line contains N space separated integers $1 \leq A_1, \dots, A_N \leq 10^6$ denoting the numbers written on Rami's cards
- The third line contains N space separated integers $1 \leq B_1, \dots, B_N \leq 10^6$ denoting the numbers written on Yessine's cards.

Output

Rami if Rami will win. Otherwise **Yessine** if he will win.

Examples

standard input	standard output
3 3 6 3 4 1 2 3	Rami
5 7 2 3 4 5 6 2 3 5 7 2	Yessine

Note

- It is guaranteed that the game will always have a winner.
- For the first test case, **Rami** can force a win by choosing 3, 2, 3 on this order.
- For the second test case, whatever **Rami** will play on his first turn, **Yessine** can win instantly by choosing 7

Problem H1. Rami's Scheme (Easy Version)

Input file: standard input
Output file: standard output
Time limit: 6 seconds
Memory limit: 512 megabytes

This is the easy version of the problem. The difference between the two versions are the constraints.

Rami has been always fond of random numbers, he keeps wondering about how randomness arises from the deterministic nature of mathematics.

Wanting to impress his friends, he created a new pseudo-random number generation scheme, that he proudly called Rami Scheme.

A Rami Scheme is used to generate s random numbers, it consists of the following steps:

1. choose 3 integer parameters: p, a, b such that $0 \leq a, b, < p$ with p prime
2. choose 2 seeds u_0, u_1
3. for $k > 1$, u_k will be generated with the following rule:

$$u_k = (au_{k-1} + bu_{k-2}) \bmod p$$

4. using the rule above, he will calculate many such numbers and use them to generate the following random numbers $(w_k)_{k \in \mathbb{N}}$:

$$w_k = \left(\sum_{i=0}^k i u_i \right) \bmod p$$

5. Finally, he will choose s numbers $w_{n_1}, \dots, w_{n_s}$. those final numbers will be his sought random numbers.

Rami wants you to evaluate his scheme:

- First of all, he wants you to measure the robustness index R of this scheme, which is defined as the **eventual fundamental period** of the sequence $(w_k)_{k \in \mathbb{N}}$. In other words, he wants the **smallest strictly positive** integer R such that:

$$\exists K \in \mathbb{N} / \quad \forall k \in \mathbb{N}_{\geq K}, w_{k+R} = w_k$$

- After that, he knows that he cannot calculate all terms of the sequence $(w_k)_{k \in \mathbb{N}}$, and he only needs s terms $w_{n_1}, \dots, w_{n_s}$ of it. So he asks your help for it.

As the number R may be too big, Rami will be happy if you only give him $R \bmod 2^{64}$.

Input

1. The first line contains 3 integers p, a, b : the parameters of the scheme, with $0 \leq a, b < p < 100$ and p a prime number.
2. The second line contains 2 integers $0 \leq u_0, u_1 < p$: the seeds.
3. The third line contains one integer $0 \leq s \leq 10^6$: the number of terms to calculate.
4. The final line contains s integers $0 \leq n_1, \dots, n_s \leq 10^6$ representing the terms to calculate.

Output

The first line contains one integer R : the robustness index of the selected scheme, modulo 2^{64} .

The second line contains s integers $v_{n_1}, \dots, v_{n_s}$: the calculated terms.

Examples

standard input	standard output
97 3 5 0 1 3 2 7 3	912576 7 79 49
17 0 0 2 3 7 0 1 2 3 4 5 6	1 0 3 3 3 3 3 3

Note

- In this version, it's guaranteed that $R < 10^6$
- It is also guaranteed that the sequence $(v_n)_{n \in \mathbb{N}}$ is eventually periodic

Problem H2. Rami's Scheme (Hard Version)

Input file: standard input
Output file: standard output
Time limit: 15 seconds
Memory limit: 1024 megabytes

This is the hard version of the problem. The difference between the two versions are the constraints.

Rami has been always fond of random numbers, he keeps wondering about how randomness arises from the deterministic nature of mathematics.

Wanting to impress his friends, he created a new pseudo-random number generation scheme, that he proudly called Rami Scheme.

A Rami Scheme is used to generate s random numbers, it consists of the following steps:

1. choose 3 integer parameters: p, a, b such that $0 \leq a, b, < p$ with p prime
2. choose 2 seeds u_0, u_1
3. for $k > 1$, u_k will be generated with the following rule:

$$u_k = (au_{k-1} + bu_{k-2}) \bmod p$$

4. using the rule above, he will calculate many such numbers and use them to generate the following random numbers $(w_k)_{k \in \mathbb{N}}$:

$$w_k = \left(\sum_{i=0}^k i u_i \right) \bmod p$$

5. Finally, he will choose s numbers $w_{n_1}, \dots, w_{n_s}$. those final numbers will be his sought random numbers.

Rami wants you to evaluate his scheme:

- First of all, he wants you to measure the robustness index R of this scheme, which is defined as the **eventual fundamental period** of the sequence $(w_k)_{k \in \mathbb{N}}$. In other words, he wants the **smallest strictly positive** integer R such that:

$$\exists K \in \mathbb{N} / \quad \forall k \in \mathbb{N}_{\geq K}, w_{k+R} = w_k$$

- After that, he knows that he cannot calculate all terms of the sequence $(w_k)_{k \in \mathbb{N}}$, and he only needs s terms $w_{n_1}, \dots, w_{n_s}$ of it. So he asks your help for it.

As the number R may be too big, Rami will be happy if you only give him $R \bmod 2^{64}$.

Input

1. The first line contains 3 integers p, a, b : the parameters of the scheme, with $0 \leq a, b < p < 10^9$ and p a prime number.
2. The second line contains 2 integers $0 \leq u_0, u_1 < p$: the seeds.
3. The third line contains one integer $0 \leq s \leq 10^6$: the number of terms to calculate.
4. The final line contains s integers $0 \leq n_1, \dots, n_s \leq 10^{18}$ representing the terms to calculate.

Output

1. The first line contains one integer R : the robustness index of the selected scheme, modulo 2^{64} .
2. The second line contains s integers $w_{n_1}, \dots, w_{n_s}$: the calculated terms.

Examples

standard input	standard output
7 2 6 0 1 14 0 1 2 3 4 5 6 7 8 9 10 11 12 13	7 0 1 5 0 2 6 0 0 1 5 0 2 6 0
499 13 41 0 15 5 0 1 2 3 4	31062750 0 15 405 374 47
17 0 0 2 3 7 0 1 2 3 4 5 6	1 0 3 3 3 3 3 3
5 0 0 4 0 8 0 1 2 3 4 5 6 7	1 0 0 0 0 0 0 0 0
975151579 1 1 0 1 5 0 1 2 3 4	950920601051041662 0 1 3 9 21

Note

- A period T of a sequence $(H_n)_{n \in \mathbb{N}}$ is a strictly positive integer such that $\forall k \in \mathbb{N}, H_{T+k} = H_k$.
- A sequence is said to be periodic if it has at least one period. The smallest such integer is called the **fundamental period**.
- An eventual period T of a sequence $(H)_{n \in \mathbb{N}}$ is an integer for which there exists certain integer K that $H_K, H_{K+1}, \dots$ is periodic with a period T .
- a sequence $(H)_{n \in \mathbb{N}}$ is said to be eventually periodic if it has at least one eventual period. The smallest such eventual period is called **the eventual fundamental period**.
- It is guaranteed that the sequence $(w_n)_{n \in \mathbb{N}}$ is eventually periodic.
- For the first test case, the first 14 terms of the sequence $(u_n)_{n \in \mathbb{N}}$ are: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13. Therefore, the first 14 terms of the sequence $(w_n)_{n \in \mathbb{N}}$ would be: 0, 1, 5, 0, 2, 6, 0, 0, 1, 5, 0, 2, 6, 0. It can be proven that $(w_n)_{n \in \mathbb{N}}$ is indeed periodic as it appears, and its fundamental period is 7.
- For the second test case, the first 5 terms of the sequence $(u_n)_{n \in \mathbb{N}}$ are: 0, 15, 195, 156, 43. Clearly, $w_0 = 0$, $w_1 = 15$, we have: $w_2 = (15 + 2 \cdot 195) \bmod 499 = 405$, $w_3 = (405 + 3 \cdot 156) \bmod 499 = 374$, and $w_4 = (374 + 4 \cdot 43) \bmod 499 = 47$. It can be proven that $(w_n)_{n \in \mathbb{N}}$ is periodic with a fundamental period of 31062750.
- For the third test case, It can be proven that $u_n = 0 \quad \forall n \geq 2$. Therefore, $w_n = 3 \quad \forall n \geq 1$. As $w_0 = 0$, the sequence $(w_n)_{n \in \mathbb{N}}$ is periodic starting from $N = 1$. Its fundamental period is 1.

- For the fourth test case, It can be proven that $w_n = 0 \quad \forall n \in \mathbb{N}$. Therefore, $(w_n)_{n \in \mathbb{N}}$ is periodic starting from $N = 0$ with a fundamental period of 1

Problem I. Non-Increasing Dilemma

Input file: **standard input**
Output file: **standard output**
Time limit: 1 second
Memory limit: 256 megabytes

Oussama, Rami and Yessine were taking a walk at late night when they found an oil lamp. When they wiped it, a genie appeared and told them they have 3 wishes, one for each of them. Oussama took the first wish. he said: "I wish I had an infinite amount of money". The genie granted him his wish and Oussama became the world's richest person.

Rami took the second wish. He said: "I wish to become a famous mathematician". The genie granted him his wish, Rami proved the Riemann hypothesis and became the most famous mathematician in his era.

Now, it is Yessine's turn to take the last wish. He said: "I wish I could become a legendary grandmaster". However, the genie said : "that is beyond my reach".

- "but why?", Yessine asked.

- "to become a legendary grandmaster is the deepest desire of every human, it is the hardest wish to accomplish and I cannot grant it", answered the genie.

Yessine laid down and got depressed. The genie felt sorry for him. He said:

- "There is a way to become one, but it is so risky".

- "I am listening", said Yessine confidently.

- "In order to become a legendary grandmaster, you need to prove yourself worthy. I will give a very hard problem, if you happen to solve it, your wish will be granted. If you fail however, I will have to take your soul."

Yessine accepted the challenge.

The genie said: "I shall give you an array $a_1, a_2, \dots, a_n$. You have to perform this operation on the array any number of times :

Choose an integer $i \in \{1, \dots, n\}$ and for all $j \neq i$ add $a[i]$ to $a[j]$

What is the minimum number of operations needed to make the array sorted in non-increasing order?"

Yessine found himself incapable of solving the problem. He desperately needs your help. Please save Yessine's soul.

Input

The first line contains a single integer t ($1 \leq t \leq 10^5$) : the number of test cases.

1. The first line of each test case contains a single integer $1 \leq n \leq 10^5$: the length of the array.
2. The second line of each test case contains n integers $1 \leq a_i \leq 10^9$: the elements of the array.

It is guaranteed that the sum of n over all test cases does not exceed 2×10^5 .

Output

For each test case, print one integer k : The minimum number of operations needed to make the array sorted in non-increasing order.

Example

standard input	standard output
5	2
4	0
4 1 2 3	3
4	0
4 3 3 2	1
4	
1 2 3 4	
1	
1	
2	
1 2	

Note

In the first test case, in the first operation, we choose $i = 3$, the array becomes $[6, 3, 2, 5]$. In the second operation, we choose $i = 4$, the array becomes $[11, 8, 7, 5]$.

Problem J. Journey Through Time

Input file: `standard input`
Output file: `standard output`
Time limit: 1 second
Memory limit: 256 megabytes

Oussama is a student. Next year will be his last one at university so he will need to find a graduation internship. He wants to find an internship abroad but he is afraid he won't be able to travel due to the wide spread of COVID19, so he started thinking about a way to end the pandemic. Suddenly, a very good idea hit him : he will make a time machine and go back in time to find the very first infected persons and save them from the infection thus preventing the pandemic. However, time travel is dangerous and can cause some serious damage.

You might have heard that the timeline is linear, but it is not. In fact, the timeline is a tree of N nodes, where each node is a different time. Each node i causes a certain damage D_i .

There are M special nodes. Each special node denotes a time of an infection that should be stopped. Oussama is initially at the present, which is node 1, and will visit each one of the special nodes in any order he wants. Whenever he moves from a node A to a node B , he will get a damage equal to the **maximum node damage** on the **path** between A and B . However, he won't get any more damage from the nodes on that path after the first time he travels through it. In other words, after he walks through a path and takes the damage, the damage of the nodes on that path **become zeros**. When he visits the node he wants, he stops to prevent the infection from happening and then travel from there to one of the remaining special nodes. He repeats the same process until every special node is visited . The total damage he will get is the **sum of the damages** obtained on each move.

Oussama wants to save the world from the pandemic but he has a limited stamina so he can't take too much damage. Help him determine the **minimal damage** he will take during the time travel .

Input

The first line contains 2 integers N and M ($1 \leq M \leq N \leq 2 \cdot 10^5$)

The second line contains M space separated integers denoting the indices of special nodes.

The third line contains N space separated integers denoting the damage of each node ($0 \leq D_i \leq 10^9$).

Finally, there are $N - 1$ lines, with each containing 2 space separated integers U, V ($1 \leq U, V \leq N$) indicating that there is an edge between nodes U and V .

Output

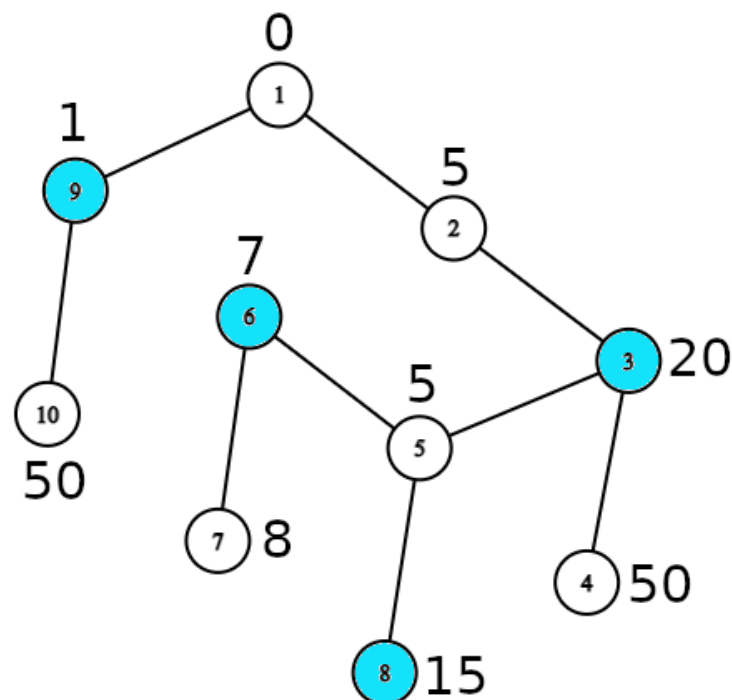
Print a single integer, the minimal total damage.

Examples

standard input	standard output
<pre> 10 4 3 6 8 9 0 5 20 50 5 7 8 15 1 50 1 2 1 9 9 10 2 3 3 4 3 5 5 6 5 8 6 7 </pre>	28
<pre> 10 5 4 6 7 9 10 0 1 1 3 8 5 4 11 9 10 1 2 1 3 2 4 2 5 5 9 5 10 3 6 6 7 7 8 </pre>	27

Note

For the first test case, the representation of the graph is:



Problem K. Count the squares

Input file: `standard input`
Output file: `standard output`
Time limit: 0.5 seconds
Memory limit: 12 megabytes

You are given a rectangle of dimensions $N \times M$ composed of 1×1 squares. Your job is to count the number of squares inside that rectangle. The squares we are talking about can have any side length.

For example the number of squares inside a 2×3 rectangle is 8 : 6 of side length 1 and 2 of side length 2.

Input

The first and only line of the input contains two integers $1 \leq N, M \leq 10^6$

Output

Print a single integer, the answer to the problem

Examples

standard input	standard output
2 3	8
4 4	30

Problem L. Two Shortest Paths

Input file: standard input
Output file: standard output
Time limit: 1 second
Memory limit: 256 megabytes

Yessine and Rami are living in Sfax.

Sfax is a rectangular grid of size $n \times m$. Each cell has a cost. The cost of the cell (i, j) is $a_{i,j}$.

Initially, All the costs are unset.

Yessine is living in cell $(1, 1)$ and wants to go to the cell (n, m) , and Rami is living in cell $(1, m)$ and wants to go to the cell $(n, 1)$.

Rami and Yessine can move to any other cell that share the same edge with their current cell, in other words they can move **UP**, **DOWN**, **LEFT**, or **RIGHT**.

Between all the paths, to reach their destinations, Yessine will take a path that has a minimum cost. Also, Rami will take independently of Yessine a path that has a minimum cost.

The cost of a path is the sum of all numbers in all cells in this path including the start cell and the end cell.

As Oussama is their best friend, he wanted to put on each cell (i, j) **distinct** strictly positive integers so that the **sum** of the two costs of paths of Yessine and Rami is as **minimal** as possible.

Oussama is stuck. Please help him determine what is the **minimal sum** of the two costs.

Input

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

Each test case contains a line of two integers n and m denoting the size of the grid ($2 \leq n, m \leq 10^4$)

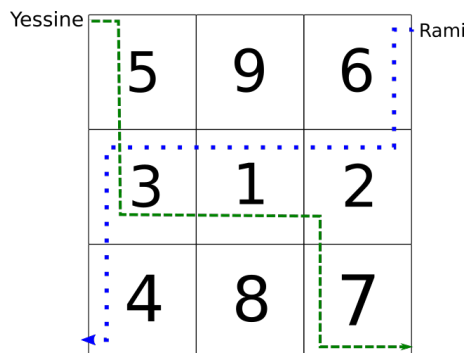
Output

For each test case, print a single integer C — The minimal sum of the two costs.

Example

standard input	standard output
3	34
3 3	81
5 4	94
7 3	

Note



For the first test case, the following grid describes a possible trajectory taken by Rami and Yessine with a minimal sum of 34.

Problem M. Mean Absolute Deviation

Input file: standard input
Output file: standard output
Time limit: 2 seconds
Memory limit: 256 megabytes

Rami loves many branches of mathematics. But, if there is an exception, it must be statistics.

For usual mathematical problems, he will devote his time to understand the structure of the problem and gain intuition, but for statistics, he will forward it to a friend.

But when Statistics and Competitive Programming intersects, that friend should be Yessine who received a recent email from Rami:

Dear Yessine

Today, I have received an interesting problem that I hope that you will like.

Spoiler Alert: This problem is difficult, even for you ☺:

Given n reals $x = [x_1, \dots, x_n]$, I challenge you to calculate the mean absolute deviation $\delta(x)$ (MAD):

$$\delta(x) = \frac{1}{n} \sum_{i=1}^n |x_i - \mu(x)|$$

$$\text{where: } \mu(x) = \frac{1}{n} \sum_{i=1}^n x_i \text{ is the mean}$$

Now, to make things worse, I challenge you to redo the calculation for q subarrays of $x : x_{[l_1, r_1]}, \dots, x_{[l_q, r_q]}$:

Find $\delta(x_{[l_i, r_i]})$ for each $i \in \{1, \dots, q\}$

Here is the list of constraints:

$1 \leq n \leq 10^5$	$1 \leq q \leq 10^5$	$1 \leq l_i \leq r_i \leq n$
----------------------	----------------------	------------------------------

Best Regards,

Rami.

Yessine was mad that such an easy problem was given to him, or so he thought until reading the constraints.

Now, neither Rami nor Yessine are capable of solving this problem, so they both request your help.

Input

1. The first line contains two integers $1 \leq n, q \leq 10^5$ representing the size of the array and the number of queries
2. The second line contains n reals $x_1, \dots, x_n$ representing the values of the array $1 \leq x_i \leq 10^9$
3. Each of the following q lines contains 2 integers l, r with $1 \leq l, r \leq n$ representing the considered subarray $x_{[l, r]}$

Output

q reals, with the i^{th} line representing the mean absolute deviation of the subarray $x_{[l_i, r_i]}$

Examples

standard input	standard output
5 7 0 10 0 1 3 1 3 2 4 2 2 4 5 3 5 1 2 3 4	4.444444 4.222222 0.000000 1.000000 1.111111 5.000000 0.500000
5 7 4 3 0 5 8 3 4 2 2 1 4 3 5 2 3 2 3 3 4	2.500000 0.000000 1.500000 2.888889 1.500000 1.500000 2.500000

Note

Yessine received a clarifying mail from Rami:

To Yessine

Well, I accidentally omitted the definition of the mean absolute deviation of a subarray, here is it with a little example:

- The mean of the subarray $x_{[l_i, r_i]}$ is :

$$\mu(x_{[l_i, r_i]}) = \frac{1}{r_i - l_i + 1} \sum_{j=l_i}^{r_i} x_j$$

- The mean absolute deviation of the subarray $x_{[l_i, r_i]}$ is :

$$\delta(x_{[l_i, r_i]}) = \frac{1}{r_i - l_i + 1} \sum_{j=l_i}^{r_i} |x_j - \mu(x_{[l_i, r_i]})|$$

- For a little example, consider the array $x = [0, 10, 0, 1, 3]$ and the query $l = 1, r = 3$. We have $x_{[1, 3]} = [0, 10, 0]$. The mean of the subarray is $\mu(x_{[1, 3]}) = \frac{10}{3}$ and the MAD is:

$$\delta(x_{[1, 3]}) = \frac{1}{3} \sum_{j=1}^3 |x_j - \mu(x_{[1, 3]})| = \frac{|0 - \frac{10}{3}| + |10 - \frac{10}{3}| + |0 - \frac{10}{3}|}{3} = \frac{\frac{10}{3} + \frac{20}{3} + \frac{10}{3}}{3} = \frac{40}{9} \approx 4.44444$$

Good Luck,

Rami.