The International Collegiate Programming Contest
Sponsored by ICPC Foundation

The 2021 Angolan Collegiate Programming Contest

(Contest Problems)

Angola
November 2021

Problem A. Look Carefully

Input file:	`chi.in`
Output file:	`standard output`
Balloon Color:	`Pink`

Sevawy doesn't want to waste your time reading a long legend story. So, he will go straight to the task he made specially for you.

You have this word "chi". *Sevawy* will give you a character and ask if the character is in this word or not.

Input

The first and only line contains a lowercase English character c

Output

You should output "Yes"if the character exists in the word. Otherwise, you should output "No".

You can print "Yes"or "No"in any case (upper or lower).

Example

chi.in	standard output
c	Yes

Problem B. Eslam and the prime numbers

Input file:	`primes.in`
Output file:	`standard output`
Balloon Color:	`Orange`

Eslam loves playing with prime numbers that are digits yet. These numbers are **2, 3, and 5**. But Eslam grew rather bored of such numbers, so he came up with a few games that were connected with them.

Eslam wants to find the minimum number of length N, such that it is simultaneously divisible by all numbers Eslam already knows (2, 3, and 5).

A number's length is the number of digits in its decimal representation without leading zeros.

Input

The first line contains a single integer T ($1 \leq T \leq 10^3$), denoting the number of test cases.

Each test case contains one integer N ($1 \leq N \leq 10^4$).

Output

For each test case print a single line containing one integer the answer to the problem without leading zeroes, or -1 if the number that meets the problem condition does not exist.

Example

primes.in	standard output
1 3	120

Note

A leading zero is any 0 digit that comes before the first nonzero digit in a number

Example:- 003, 01, 0100 consider leading zero numers but 300, 10, 100, 101 not leading zero numbers

Problem C. Rotten Apples

Input file:	`apples.in`
Output file:	`standard output`
Balloon Color:	`Yellow`

It's apple season and Ahmed is arranging his apple harvest into boxes to send to traders. Each box is a rectangular grid of size $n \times m$ (n rows by m columns).

Unfortunately, Ahmed found x rotten apples in his harvest. Instead of throwing them away, he (immorally) decided to put them in a box with the rest of the good apples so that he can complete an extra box to send to traders to gain more money. He knows that each rotten apple will turn all the apples that are directly adjacent to it rotten (spoil them). Here, directly adjacent means above, below, to the left or to the right, but not diagonally adjacent.

Despite committing this immoral act, Ahmed wants to minimize the number of apples that will turn rotten. Help him place the rotten apples in such a way that the number of good apples that will turn rotten is minimum.

Input

The first line containts t, the number of test cases.

Each of the next t lines contain three integers, n, m, and x, $(2 \leq n, m \leq 2{\times}10^5)$, $(1 \leq x \leq n{\times}m \leq 2{\times}10^5)$, the height and width of the box, and the number of rotten apples.

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

Output

Output the contents of the box after placing the rotten apples optimally.Print n lines each consisting of m elements without any spaces. Use '.' in place of a good apple and '*' in place of a rotten apple. Print the contents before the rotten apples spoil the good apples.

Example

apples.in	standard output
1 2 3 2	*.. *..

Problem D. Make it arithmetic progression

Input file:	`make.in`
Output file:	`standard output`
Balloon Color:	`Light Green`

Nihad Told Jasper "Less talking, more looking, OK, Jasper?"

So Jasper wrote this problem in Nihad's way:

You are given an array A with n numbers in such away that for every $1 < i$ then $A_{i-1} \leq A_i$.

You can do the following operation m times:

- Choose any number x and insert it in any position you want.

So after do that the size of array A become $n + m$.

The new array should become arithmetic progression (i.e. for every $2 < i$ then $A_i - A_{i-1}$ equal to $A_2 - A_1$).

You have to count number of ways to make the array like arithmetic progression, two ways are considered different if elements in the resulting arrays are different.

For example if $A = [2, 6, 8]$ and $m = 4$, we can make some arrays:

$A = [-2, 0, 2, 4, 6, 8, 10]$

$A = [2, 4, 6, 8, 10, 12, 14]$

$A = [2, 3, 4, 5, 6, 7, 8]$

Input

$T(T \leq 100)$ number of test cases.

The first line of each test you are given n, m $(2 \leq n \leq 10^5$, $0 \leq m \leq 10^9)$.

The second line of each test you are given n numbers $A_i(0 \leq A_i \leq 10^9)$ in ascending order.

Sum n over all tests dose not exceed 3×10^5.

Output

Number of ways to make array A arithmetic progression.

Example

make.in	standard output
1 3 4 2 6 8	5

Problem E. Yasser's Permutation

Input file:	lettercasepermutation.in
Output file:	standard output
Balloon Color:	White

Yasser has a string X the password of the spacecraft, they will play a game to change this password as follow:

- changes any lowercase letter in X to be uppercase and vice-versa.

Print all possible passwords they could obtain in lexicographical order.

Input

Only one line contains string $s(1 \leq |X| \leq 12)$ consists of lowercase or uppercase English letters and digits.

Output

Print all possible passwords they could obtain in lexicographical order.

Examples

lettercasepermutation.in	standard output
nGu	NGU NGu NgU Ngu nGU nGu ngU ngu
ya55er	YA55ER YA55Er YA55eR YA55er Ya55ER Ya55Er Ya55eR Ya55er yA55ER yA55Er yA55eR yA55er ya55ER ya55Er ya55eR ya55er

Problem F. Strings wings

Input file:	`string.in`
Output file:	`standard output`
Balloon Color:	`Dark Blue`

Yasser loves strings :)

Given an integer N and two strings s and t, Yasser loves strings that satisfy the following conditions:

1 - The first N characters equal to the string s.

2 - The last N characters equal to the string t.

3 - The length of the string is at least N.

Find the minimum length of the string that satisfies the conditions.

Input

The first line of the input contains one number N $(1 \leq N \leq 10^3)$.

The second line of the input contains string s $(|s| = N)$.

The third line of the input contains string t $(|t| = N)$.

Output

Print length of the string that satisfies the conditions.

Example

string.in	standard output
3 abc cde	5

Note

In the example the minimum length is 5 because the string is "abcde".

The first N characters equal to the string s = abc.

The last N characters equal to the string t = cde.

Problem G. parallel universe

Input file:	`parallel.in`
Output file:	`standard output`
Balloon Color:	`Silver`

In a parallel universe, there was a small country, it had great economical growth, and because of that, a lot of people from different nations migrate into it.

After a period of time, the residents noticed that the arrangement of their houses was random and they would be happier if all the people who spoke the same language lived as close as possible to each other.

A new government became in charge, and it wanted to solve that problem by making Everyone who speaks the same language live beside each other.

There are N houses and K different languages, the language of the owners of the houses can be represented as an array of N elements $a_1, a_2, ..a_n$ such that each element represent the language of the corresponding house.

And in order to do that, it made a new law, allowing it to move any two adjacent houses, in other words, the government can swap house number i with house number $i+1$ or house number $i-1$ (if it does exists).

But the government doesn't want to make any unnecessary moves because it's not cheap to move people around. so they want your help to determine the minimum swaps needed in order To achieve the optimal arrangement of the houses of the residents.

for example: $a = [1, 2, 3, 1, 2, 3]$

1^{st} swap between 3 and 4 will make the array: $a = [1, 2, 1, 3, 2, 3]$

2^{nd} swap between 2 and 3 will make the array: $a = [1, 1, 2, 3, 2, 3]$

3^{rd} swap between 4 and 5 will make the array: $a = [1, 1, 2, 2, 3, 3]$

so the answer is 3

Input

The input file starts with a single integer T the number of testcases. Each of the test cases starts with 2 integers ($N \leq 10^5$) the number of houses, and ($K \leq 20$) the number of different languages. the next line will have n integers ($a_i \leq K$) the languages of the house i-th owners.

Output

An integer representing the minimum number of swaps needed.

Example

parallel.in	standard output
1 6 3 1 2 3 1 2 3	3

Problem H. play with prime

Input file:	play.in
Output file:	standard output
Balloon Color:	Purple

One day our friend Mohammad was bored, so he called his friend Ayham to play with him a nice game.

The game starts with an array of N numbers, each number A[i] is the product of all prime numbers between L[i] and R[i] inclusive.

Each turn a player will choose a prime number X ($X > 1$) and remove all numbers where X is a divisor to those numbers.

For example if the array = [3, 15, 7, 11] and the player choose a number 3 then the array will be = [7, 11]. (note : every numberin the array is a result of the product of some prime numbers, for instance, 15 = 5 * 3).

The game ends if a player can't choose any number (in other words, if the array is empty).

Since Ayham is an electrical engineer Mohammad will play first. Help our friend to find out the winner if both players played optimally.

Input

The first line contains an integer N ($1 \leq N \leq 10^5$) the number of elements in the array.

The next N lines contain two integers L[i] and R[i] ($1 \leq L[i] \leq R[i] \leq 10^7$) , where A[i] is equal to product of all prime numbers between L[i] and R[i] inclusive, it's guaranteed that for each $L[i] < L[j]$ then $R[i] < L[j]$, in other words, if you sort all pair according to their L[i] in an ascending order, each L[j] will be greater than all its previous R[i].

Output

print "Ayham" if the winner is Ayham otherwise print "Mohammad".

Example

play.in	standard output
2 3 2 3 8 12 14 17 3 2 2 8 10 14 17	Ayham Mohammad

Problem I. Akrik Country

Input file:	`akrik.in`
Output file:	`standard output`
Balloon Color:	`Red`

Akrik is a strange country. All people in Akrik hate loneliness.

There are n cities in Akrik, numbered from 1 to n, the capital is city number 1, and each city contains a_i citizens where $(1 \leq i \leq n)$. The cities of Akrik form a tree.

Because all Akrik people like to socialize, Akrik's government plans to organize a great party in the capital to make all citizens have fun. Citizens walk from the city they are in to another city that is closer to the capital (citizens can't walk to a city after reaching the capital). Akrik's people don't like walking alone so they decide to gather in groups of **exactly** k citizen. Every such group will go directly to the capital. If a citizen can't find a group to go with, he will walk alone to another city make one step.

You are asked to find the minimum total number of steps that Akrik citizens must walk alone to reach the capital.

Input

The first line contains a single integer T $(1 \leq T \leq 50)$ — the number of test cases. The description of the test cases follows.

The first line of each test case contains two integers n, k $(1 \leq n, k \leq 10^5)$ denote the number of cities in Akrik and the number of citizens required for each group respectively.

The second line of each test case contains $n-1$ integers where the i_{th} integer $x(1 \leq x \leq i)$ means that there is a road between the cities x and $i+1$.

The third line of each test case contains n integers where the i_{th} integer $a_i(1 \leq a_i \leq 10^5)$ denotes the number of citizens in the city i.

Output

Print one integer denotes the minimum sum of distances that Akrik citizens must walk alone to reach the capital.

Example

akrik.in	standard output
3 3 3 1 2 7 3 3 3 3 1 1 3 7 6 4 6 1 1 2 12 2 5 3	0 1 13

Problem J. XOR MST

Input file:	xormst.in
Output file:	standard output
Balloon Color:	Dark Green

given a complete undirected weighted graph with n vertices.

The i_{th} vertex has number i on it, and the weight of an edge between vertices i and j is equal to i *xor* j.

Calculate the weight of the minimum spanning tree in this graph.

Input

The first and only line of input contains one integer n ($2 \leq n \leq 10^{14}$)

Output

Print the answer to the problem .

Example

xormst.in	standard output
3 5 3 1598753	8 3 17239408

Problem K. Lazy Busher

Input file:	`plan.in`
Output file:	`standard output`
Balloon Color:	`Black`

Back in the days, Busher was a very motivated competitive programmer. He used to always have a plan for the next k days. The plan consists of n practice sessions, each session takes place on someday between 0 and $k-1$. Also, there might be more than one session that takes place on the same day. Busher programmed this plan in an app on his mobile phone.

The app will simply list what sessions Busher has for the next k days. When a day passes, the app will reschedule the same sessions of the past day, to a new day after all already planned days. In other words, the plan repeats itself every k days.

For example, a plan for the next 5 days might look like this: $P = [0, 1, 5, 0, 10]$, which means:

- Days 0 and 3, have no practise sessions.
- Days 1, 2 and 4, have 1, 5 and 10 practise sessions replectively.

Busher got lazy over time and started to skip practice. He has not trained for some time now. He is sure, that the number of skipped days is less than k, but still can't tell how many.

Luckily for him, he still remembers the plan when he last practiced. Also, he still has the app installed.

Given the plan that he last remembers, and the app's plan for today, determine the minimum possible number of days he might have skipped to practice on. If there's no such answer print -1.

Input

The first line of the input contains a single integer number T. The number of test cases. Each test case is described as follows :

The first line of each test case contains two integers n and k $(1 \leq n \leq 10^5)$ $(1 \leq k \leq 10^9)$ and the number of sessions of the plan, and how many days the plan has.

The following line contains n integers $L_1, L_2, ..., L_n$, $(0 \leq L_i < k)$, the last plan Busher remembers.

The following line contains n integers $T_1, T_2, ..., T_n$, $(0 \leq T_i < k)$. the plan of today.

It's guaranteed that the sum of n over all test cases doesn't exceed 10^6.

Output

For each test case, print a single integer, x $(0 \leq x < k)$, the minimum number of days he might have skipped. Or -1 if there's no valid answer.

Example

plan.in	standard output
4 3 5 2 3 4 4 0 1 5 10 3 0 3 2 9 2 3 9 3 0 3 10 1 2 3 4 5 7 5 10 1 3 6 7 8 9 0 1 3 5	3 0 -1 -1

Note

In the first test case, the plan for the next 5 days since the last time Busher practiced will change as follow:

- Day 0: $[0, 0, 1, 1, 1]$
- Day 1: $[0, 1, 1, 1, 0]$
- Day 2: $[1, 1, 1, 0, 0]$,
- Day 3: $[1, 1, 0, 0, 1]$, Which is the app's plan for today. So the answer is 3.

In the second test case, both plans are the same.

Problem L. Divisibility by two

Input file:	`math.in`
Output file:	`standard output`
Balloon Color:	`Gold`

Yasser loves numbers divisible by two.

You are given a positive integer N among the integers between 1 and N (inclusive), find the number that can be divisible by two for the most number of times.

Input

The only line of the input contains one number N ($1 \leq N \leq 10^{18}$) .

Output

Print the answer as described.

Examples

math.in	standard output
1	1
100	64

Note

Here, the number of times an integer can be divisible by 2, is how many times the integer can be divided by 2 without remainder.