Problem A. Airplane Cliques

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

There are n cities in Saint Waterloo. They are connected with $n - 1$ undirected airlines, such that it is possible to get from any city to any other city by using airlines.

Let's say that the two cities are in a good relationship if it is possible to get from one to another using at most x flights.

Additionally, let's say that the set of k cities is **friendly** if all pairs of cities in it are in a good relationship.

You need to find the number of friendly sets of cities for each k, such that $1 \leq k \leq n$. As these values may be very large, find them modulo $998\,244\,353$.

Input

The first line of input contains two integers n, x ($1 \leq n \leq 300\,000, 0 \leq x \leq n - 1$): the number of cities in Saint Waterloo and x.

The next $n - 1$ lines contain descriptions of edges, such that the i-th line contains two integers a, b ($1 \leq a, b \leq n$), the indices of cities connected by the i-th airline.

Output

Print n integers: the number of friendly sets of $1, 2, \ldots, n$ cities, modulo $998\,244\,353$.

Examples

standard input	standard output
1 0	1
5 1 1 2 2 3 3 4 4 5	5 4 0 0 0
4 2 1 2 1 3 1 4	4 6 4 1

Note

In the second example, all possible friendly sets are of size 1 and 2, and the number of friendly sets for these sizes is the number of cities and the number of airlines, respectively.

In the third example, it is possible to get from any city to any other city using at most x flights, so the number of friendly sets of k cities is $\binom{4}{k}$.

Problem B. Best Tree

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	512 mebibytes

You are given the degree sequence of a tree (degrees of all its vertices, in arbitrary order).

Among all trees with the given degree sequence, find a tree with the largest maximum matching.

Input

The first line of input contains one integer t ($1 \leq t \leq 100\,000$): the number of testcases.

Next lines contain t descriptions of a test case.

The first line of each test case contains one integer n ($2 \leq n \leq 200\,000$): the number of vertices.

The next line contains n integers $d_1, d_2, \ldots, d_n$ ($1 \leq d_i \leq n - 1$), the degree sequence of a tree.

It is guaranteed that $\sum d_i = 2(n-1)$ and that there is at least one tree with the given degree sequence.

Also, it is **guaranteed** that the total sum of n in all test cases is at most $200\,000$.

Output

For each test case, print one integer: the largest maximum matching among all trees with the given degree sequence.

Example

standard input	standard output
2	5
10	1
1 1 2 2 2 2 2 2 2 2	
5	
4 1 1 1 1	

Note

In the first test case, you can construct a path with 10 vertices, it will have the same degree sequence and the largest possible maximum matching.

In the second test case, the only possible tree is a star (one vertex connected with all others), and the largest matching for it is 1.

Problem C. Cells Blocking

Input file:	*standard input*
Output file:	*standard output*
Time limit:	3 seconds
Memory limit:	512 mebibytes

You are given a grid $n \times m$, and some cells are blocked.

You need to find the number of ways to block two free different cells such that there will be no path from $(1, 1)$ to (n, m) which goes down or to the right by only using free cells.

Note that it is not forbidden to block cells $(1, 1)$ and (n, m). They may be blocked initially as well.

Input

The first line contains two integers n and m ($1 \le n, m \le 3000$): number of rows and columns in the grid.

Each of the next n lines contain m characters, such that the j-th character of i-th string is equal to '.' if cell (i, j) is free and '*' if it is blocked.

Output

Print one integer: the number of ways to block two cells, such that there will be no path from $(1, 1)$ to (n, m) which goes only to the right or down by only using free cells.

Examples

standard input	standard output
3 3	17
3 3 .** .*. ...	15
3 4 **** .... ****	6

Note

In the first example, if you will block $(1, 1)$ or $(3, 3)$ and any other cell, there will be no correct path. The number of such ways is $8 + 8 - 1$.

Also, if you will block $((1, 2)$ and $(2, 1))$ or $((3, 2)$ and $(2, 3))$ there will be no correct path, so the answer is $8 + 8 - 1 + 2 = 17$.

In the second example, if you block any two cells, there will be no path, so the answer is $\binom{6}{2} = 15$.

In the third example, initially, there are no paths from $(1, 1)$ to (n, m), so after blocking any two cells there still will be no paths, so the answer is $\binom{4}{2} = 6$.

Problem D. Disjoint LIS

Input file:	*standard input*
Output file:	*standard output*
Time limit:	3 seconds
Memory limit:	512 mebibytes

Let the LIS of a permutation be the length of its longest increasing subsequence.

A permutation is **good** if it is possible to find two increasing subsequences of length LIS that do not share any common elements.

Given n, find the number of good permutations with n elements. As the answer may be large, you only need to find it modulo $998\,244\,353$.

Input

The first line of input contains one integer n ($1 \le n \le 75$): the number of elements.

Output

Output one integer: the number of good permutations with n elements, modulo $998\,244\,353$.

Example

standard input	standard output
6	132

Problem E. Easy Win

Input file:	*standard input*
Output file:	*standard output*
Time limit:	2 seconds
Memory limit:	512 mebibytes

Alice and Bob are playing a game.

They have n piles of stones, such that there are a_i $(1 \le a_i \le n)$ stones in the i-th pile.

During his/her turn, each player, starting from Alice, will pick any pile and take at least one and at most x stones from it.

The player that can't make a move on his/her turn (all piles are empty) loses.

Alice and Bob still have not decided the final value of x, so they have asked you to find out who will win if both players play optimally for all values of x, such that $1 \le x \le n$.

Input

The first line of input contains one integer n $(1 \le n \le 500\,000)$: the number of piles and the upper bound on the number of stones in piles.

The next line contains n integers, $a_1, a_2, \ldots, a_n$ $(1 \le a_i \le n)$: the number of stones in piles.

Output

Print n words, where the i-th of them is "`Alice`" if Alice will win under optimal play for $i = x$, and "`Bob`" otherwise.

Examples

standard input	standard output
6 6 6 6 6 6 6	Bob Bob Bob Bob Bob Bob
4 1 2 3 4	Bob Alice Bob Alice
5 1 2 3 2 2	Bob Alice Bob Bob Bob

Note

In the first example, independently on x, Bob may do symmetrical moves (the same move on the pile with the same number of stones), so on each turn, he definitely will have at least one available move.

Problem F. Farm of Monsters

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	512 mebibytes

There are n monsters, and the i-th monster initially has h_i health points.

Let's call a monster **alive** if his HP is strictly greater than zero.

You have an attack power of a, and your opponent has an attack power of b.

As long as one monster is still alive, you and your opponent take turns fighting monsters (beginning with you).

You are very smart, so on your turn, you can attack any monster that is alive or do nothing. If you choose to attack some monster i, the monster's health, h_i, will decrease by exactly a.

After your attack, if the monster is dead (not alive), you gain one **victory point**.

Your opponent, on the other hand, is not so smart. During his turn, he finds the monster with the smallest index that is alive and attacks it (i.e. he finds the smallest i such that $h_i > 0$ and decreases h_i exactly by b).

What is the greatest number of victory points that you can obtain?

Input

The first line contains three integers n, a, b ($1 \leq n \leq 300\,000$, $1 \leq a, b \leq 10^9$): the number of monsters and the attack powers of you and your opponent, respectively.

The second line contains n integers $h_1, h_2, \ldots, h_n$ ($1 \leq h_i \leq 10^9$): the health points of the monsters.

Output

Print one integer: the largest number of victory points that you can obtain.

Examples

standard input	standard output
3 1 1 1 1 1	2
3 1 1 2 2 2	3
10 34 100 17 27 73 17 60 12 25 53 31 46	5

Note

In the first example, on your first turn, you can kill the third monster, and on your second turn, you can kill the second monster.

In the second example, you can wait until the leftmost monster will have $h_i = 1$, and then kill it yourself.

Problem G. Giant Penguin

Input file:	*standard input*
Output file:	*standard output*
Time limit:	3 seconds
Memory limit:	512 mebibytes

Pengsoo is a well-known giant Korean penguin. He is very rude and loves singing.

This time, Pengsoo is performing graph queries.

He has a connected undirected graph, where each **vertex** lies on at most k vertex-simple cycles.

He wants to answer two types of queries.

- Mark some vertex v.

- Find the closest marked vertex to the vertex v (it is guaranteed that when the query of this type is asked, there is at least one marked vertex).

Pengsoo is very lazy, and has decided to take a nap, so he asks you to perform these queries. If you do not succeed by the time he wakes up, he will bully you, so act quickly!

Input

The first line of input contains three integers n, m, k ($1 \le n \le 100\,000, n - 1 \le m \le 200\,000, 0 \le k \le 10$): the number of vertices, edges, and largest number of vertex-simple cycles that may pass through one vertex.

The next m lines contain the description of edges. The i-th of them contain two integers u_i, v_i ($1 \le u_i, v_i \le n, u_i \ne v_i$), denoting an edge between vertices u_i and v_i.

It is guaranteed that there are no multiple edges, the graph is connected, and each vertex lies on at most k vertex-simple cycles.

The next line contains one integer q ($1 \le q \le 200\,000$): the number of queries.

The next q lines contain the description of edges. The i-th of them contain two integers t_i, v_i ($1 \le t_i \le 2, 1 \le v_i \le n$).

If $t_i = 1$, mark the vertex v_i. It is guaranteed that this vertex was not marked before.

If $t_i = 2$, find the distance to the closest marked vertex from v_i. It is guaranteed that there is at least one marked vertex.

Output

For each query with $t_i = 2$, print the distance to the closest marked vertex.

Examples

standard input	standard output
5 4 0	0
1 2	1
2 3	2
3 4	1
4 5	0
7	
1 1	
1 5	
2 1	
2 2	
2 3	
2 4	
2 5	
5 6 2	2
1 2	2
2 3	
1 3	
3 4	
4 5	
3 5	
3	
1 1	
2 4	
2 5	

Problem H. Horrible Cycles

Input file:	*standard input*
Output file:	*standard output*
Time limit:	2 seconds
Memory limit:	512 mebibytes

You are given a bipartite graph with n vertices in each part.

In this graph, each vertex from the left part is connected to some prefix of vertices from the right part. Namely, the i-th vertex in the left part is connected with vertices $1, 2, \ldots, a_i$ in the right part.

Find the number of vertex-simple cycles in this graph. Two cycles are different if there exists some edge which is present in one cycle but not in the other.

As this number may be large, find it modulo $998\,244\,353$.

Input

The first line of input contains one integer n ($1 \le n \le 5000$): the number of vertices in each part.

The next line of input contains n integers $a_1, a_2, \ldots, a_n$ ($1 \le a_i \le n$): a description of the given graph.

Output

Output one integer: the number of vertex-simple cycles in the given graph, modulo $998\,244\,353$.

Examples

standard input	standard output
1 1	0
2 2 2	1
3 3 3 2	7
10 6 6 7 7 8 8 9 9 10 10	410150080

Problem I. Ignore Submasks

Input file:	*standard input*
Output file:	*standard output*
Time limit:	1 second
Memory limit:	512 mebibytes

You are given an array of n integers, $a_1, a_2, \ldots, a_n$. Each integer is between 0 and $2^k - 1$, inclusive.

Let's say that $f(x)$ is the smallest i, such that $(a_i \& x) \neq a_i$, or 0, if there are no such i. $(a \& b)$ is the bitwise AND operation.

Find $f(0) + f(1) + \ldots + f(2^k - 1)$. As this value may be very large, find it modulo $998\,244\,353$.

Input

The first line contains two integers: n, k ($1 \leq n \leq 100, 1 \leq k \leq 60$).

The next line contains n integers: $a_1, a_2, \ldots, a_n$ ($0 \leq a_i < 2^k$).

Output

Print one integer: $f(0) + f(1) + \ldots + f(2^k - 1)$, modulo $998\,244\,353$.

Examples

standard input	standard output
2 1 0 1	2
2 2 2 1	4
5 10 389 144 883 761 556	1118

Note

In the first example, $f(0) = 2, f(1) = 0$.

In the second example, $f(0) = 1, f(1) = 1, f(2) = 2, f(3) = 0$.

Problem J. Just Counting

Input file:	*standard input*
Output file:	*standard output*
Time limit:	1 second
Memory limit:	512 mebibytes

You are given an undirected graph without loops and multiple edges.

Find the number of ways to write integers $[0; 4]$ on edges such that for each vertex, the sum of weights of edges incident to it will be equal to zero modulo five (i.e. is equal to $5k$ for some integer k).

As the answer may be very large, you only need to find it modulo $998\,244\,353$.

Input

The first line of input contains one integer t ($1 \leq t \leq 500\,000$): the number of testcases.

The next lines contain t descriptions of test cases.

The first line of each test case contains two integers n, m ($1 \leq n \leq 200\,000, 0 \leq m \leq 300\,000$): the number of vertices.

The next m lines contain descriptions of edges, where the i-th of them contains two integers a_i, b_i ($1 \leq a_i, b_i \leq n, a_i \neq b_i$), denoting an edge connecting vertices a_i and b_i in the graph.

It is guaranteed that there are no multiple edges.

It is also **guaranteed** that the total sum of $n + m$ in all test cases is at most $500\,000$.

Output

For each test case, print one integer: the number of ways to write integers $[0; 4]$ on edges such that for each vertex, the sum of weights of edges incident to it will be equal to zero modulo five (i.e. is equal to $5k$ for some integer k), modulo $998\,244\,353$.

Example

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