Problem A. Cut The Wire

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

In the country of *Infinity*, there is a strange road. This road only has a starting point, but no end. Since this road is infinite, there are also countless street lights. The street lights are numbered from 1(the starting point) to infinity. The street lights are connected by wires under a strange law:

For a street light x,

- if x is even, then x is connected with $\frac{x}{2}$ by a wire;
- if x is odd, then x and $3x+1$ is connected by a wire.

Now Kris is standing in the middle of street light n and $n+1$, and he is able to cut all wires passing by. That is, he will cut all wires connecting street lights a and b satisfying $a \le n$ and $b > n$.

Now he wonders, how many wires he will cut. Please help him calculate.

Input

This problem contains multiple test cases.

The first line contains an integer $T(1 \le T \le 10^5)$ indicating the number of test cases.

The next T lines each contains one integer $n(1 \le n \le 10^9)$.

Output

For each test case, output one line of one integer indicating the answer.

Example

standard input	standard output
2 12 60	10 50

Problem B. Time-Division Multiplexing

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

Time-division multiplexing (TDM) is a method of transmitting and receiving independent signals over a common signal path by means of synchronized switches at each end of the transmission line so that each signal appears on the line only a fraction of time in an alternating pattern. This method transmits two or more digital signals or analog signals over a common channel.

The time domain is divided into several recurrent time slots of fixed length, one for each sub-channel. A sample byte or data block of sub-channel 1 is transmitted during time slot 1, sub-channel 2 during time slot 2, etc. One TDM frame consists of one time slot per sub-channel plus a synchronization channel and sometimes error correction channel before the synchronization. After the last sub-channel, error correction, and synchronization, the cycle starts all over again with a new frame, starting with the second sample, byte or data block from sub-channel 1, etc.

There are n time slots in total now, each of which periodically transmits a string only containing lowercase letters. You need to find the shortest slot length that contains all the different letters transmitted totally.

Input

This problem contains multiple test cases.

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

For each test case, the first line contains a single integer $n (1 \le n \le 100)$. Then next n lines, each line contains a string s_i only containing lowercase letters. The length of each string is no more than 12.

It is guaranteed that the $\sum n \le 2000$.

Output

For each test case, output a single integer in a line, the answer for the test case.

Example

standard input	standard output
2 2 abc bd 2 bab bbc	4 4

Note

In the first sample, there are two time slots in total and s1 = `abc`, s2 = `bd`.

Letters transmitted on the channel are as follows: `abbdcbad....`

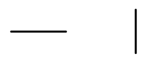
We can choose `dcba` as the answer string so the answer is 4.

Problem C. Pattern Recognition

Input file:	`standard input`
Output file:	`standard output`
Time limit:	6 seconds
Memory limit:	768 megabytes

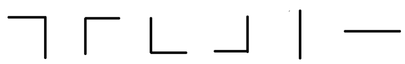
Given an image consisting only lowercase letters, for each query you should count the number of patterns in the image.

Formally, an image is a matrix with n rows and m columns, and each cell of the matrix contains a character. Considering a string consisting of lowercase letters, we can place it horizontally or vertically, so it has following shapes:



We can bend it 90 degrees, and we get some new shapes:

In a word, a pattern is a string with these 6 possible shapes:



For example, pattern **abcd** can have following shapes:

```
d         a
c         b         dcb       dc        d         d         cd        cba       bcd       ba
b         c           a        b        cba       c         b         d         a         c
a         d                    a                  ba        a                             d
```

```
abc       ab        a         a          d         a          d         a
  d        c        bcd       b          c         b        abc       dcb       dcba      abcd
           d                  cd        ab        dc
```

Kanari loves CV(Computer Vision), he writes a CNN(Convolutional Neural Network) to count the number of patterns in an image. Kanari is so cute, and the accuracy tends to 100% after a long time parameter adjustment. Can you solve it too?

Input

This problem contains multiple test cases.

The first line of the input contains an integer $T (1 \le T \le 5)$ indicating the number of test cases.

For each test case, the first line contains three integers $n, m, q (n \ge 1, m \ge 1, 1 \le n \times m \le 10^5)$, representing the width of the image, the height of the image and the number of queries.

Each of the next n lines contains m characters representing the image.

Each of the next q lines contains a string representing a query.

It is guaranteed that the sum of length of all queries in a test case is no more than 10^5.

Output

For each query, output the number of corresponding patterns occurred in the image. Note that two occurences are different if and only if the **SET** of positions for the pattern in the image are different.

Example

standard input	standard output
2 10 10 1 xxxxxxxxxx xbcdxxxxxx xaxxxdcbax xxxdxxxxxx xxxcbaxxxx xxxxxxabxx xaxaxxdcxx xbxbxxxxdc xcxcxxxxxb xdxdxxxxxa abcd 5 5 2 xsssx xxbxx xbabx xxbxx xxxxx ab sss	6 4 1

Note

Explanation for the second testcase:

For pattern **ab**, the occurences are $\{(2,3),(3,3)\},\{(3,2),(3,3)\},\{(3,3),(3,4)\},\{(3,3),(4,3)\}$.

For pattern **sss**, the only occurence is $\{(1,2),(1,3),(1,4)\}$.

Problem D. Depth First Search

Input file:	`standard input`
Output file:	`standard output`
Time limit:	6 seconds
Memory limit:	256 megabytes

Kris has a rooted tree whose root is 1. As a master of tree data structure, Kris will dfs the tree. In order to eliminate ambiguity, for each node she will visit its children from left to right. Then she defines $first(x, i)$ as the first node she visited in the subtree of x whose distance to x is i. Initially, the tree only has a root node numbered 1. However, the tree will vary over time and you should maintain Q opeartions, and each operation has one of the following format:

- 1 x y z $(2 \le x \le Q+1)$, add a leaf node numbered x as a child of y and **right** subling of z. If $z = 0$, you should regard x as the first child of y. It is guaranteed that x is unique in every operation and z is a child of y at this moment.
- 2 x $(x \ne 1)$, remove a leaf node numbered x. It is guaranteed that x doesn't have any child at this moment.
- 3 x d $(d \ge 0)$, Let's define $A = \sum_{i=0}^{d} first(x, i)$, $B = \max_{i=0}^{d}\{first(x, i)\}$. You should output A and B seperated by space. It is guaranteed that d is no more than the height of x's subtree (There exists at least one node in the subtree of x whose distance to x is d).

Input

This problem contains multiple test cases.

The first line of the input contains an integer T $(1 \le T \le 10)$, representing the number of testcases.

For each testcase, the first line contains an integer Q $(1 \le Q \le 2 \times 10^5)$, representing the number of operations.

Then Q lines follow, each line represents an operation. And the operations will be encrypted. You need to decode the operations as follows, where **key** denotes the value of $A \bmod B$ of the last type 3 operation and is initially zero for each test case:

- For type 1 operation, let $x = x \oplus key, y = y \oplus key, z = z \oplus key$.
- For type 2 operation, let $x = x \oplus key$.
- For type 3 operation, let $x = x \oplus key, d = d \oplus key$. Then you should calculate and output $A = \sum_{i=0}^{d} first(x, i)$, $B = \max_{i=0}^{d}\{first(x, i)\}$, after that set $key = A \bmod B$.

It is guaranteed that $\sum Q \le 1.5 \times 10^6$.

Output

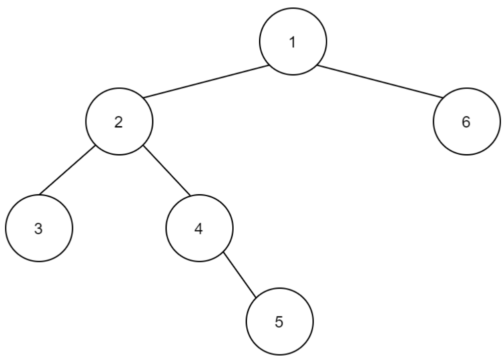
For each operation of type 3, output two integers seperated by space indicating the answer.

Example

standard input	standard output
1 9 1 2 1 0 1 3 2 0 1 4 2 3 1 5 4 0 1 6 1 2 3 1 3 3 7 1 2 3 3 1 3	11 5 6 6 12 5

Note

After the first 5 operations, the tree will look like this:



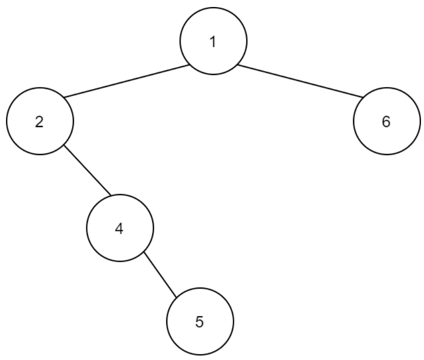
When we dfs the tree, we will visit nodes in the following order:

$$1, 2, 3, 4, 5, 6$$

For the 6th operation, the answer is $A = \sum_{i=0}^{3} first(1, i) = 1 + 2 + 3 + 5 = 11$ and $B = \max_{i=0}^{3}\{first(1, i)\}$ $= \max\{1, 2, 3, 5\} = 5$.

For the 7th operation, the answer is $A = 6$, $B = 6$.

After the 8th operation, the node 3 is removed and the new tree will look like this:



Then, for the last operation, the answer is $A = \sum_{i=0}^{3} first(1, i) = 1+2+4+5 = 12$ and $B = \max_{i=0}^{3}\{first(1, i)\}$ $= \max\{1, 2, 4, 5\} = 5$.

Problem E. Easy Math Problem

Input file:	`standard input`
Output file:	`standard output`
Time limit:	2.5 seconds
Memory limit:	256 megabytes

You need to calculate $\sum_{i=1}^{n}\sum_{j=1}^{n}\binom{i+j}{i}\cdot f(i+j,i)$, where $f(0,x)=0, f(1,x)=a$, and for all $2\le m\le x$, $f(m,x)=b\cdot f(m-1,x)+c\cdot f(m-2,x)$, and moreover, for all $m>x$, $f(m,x)=d\cdot f(m-1,x)+e\cdot f(m-2,x)$.

In mathematics, the binomial coefficients are the positive integers that occur as coefficients in the binomial theorem. Commonly, a binomial coefficient is indexed by a pair of integers $n\ge k\ge 0$ and is written $\binom{n}{k}$. It is the coefficient of the x^k term in the polynomial expansion of the binomial power $(1+x)^n$, and is given by the formula

$$\binom{n}{k}=\frac{n!}{k!(n-k)!}$$

Input

This problem contains multiple test cases.

The first line contains an integer $T(1\le T\le 50)$ indicating the number of test cases.

The next T lines each contains six integers $n,a,b,c,d,e(1\le n\le 10^5, 1\le a,b,c,d,e\le 10^6)$.

Output

Output T lines, each line contains an integer indicating the answer.

Since the answer can be very large, you only need to output the answer modulo 998244353.

Example

standard input	standard output
2 3 1 1 1 1 1 2 1 2 3 4 5	316 482

Problem F. Power Sum

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

Given a positive number n, Kris needs to find a positive number k and an array $\{a_i\}(a_i \in \{-1, 1\})$ of length $k(1 \le k \le n+2)$, such that:

$$\sum_{i=1}^{k} a_i \times i^2 = n$$

This is too hard for Kris so you have to help him.

Input

The input contains multiple test cases.

The first line contains an integer $T(1 \le T \le 100)$ indicating the number of test cases.

Each of the next T lines contains one integer $n(1 \le n \le 10^6)$.

It's guaranteed that $\sum n \le 3 * 10^7$.

Output

The output should contain $2T$ lines. For each test case, output two lines.

The first line contains one integer, k.

The second line contains a 01-string of length k representing the array, with `0` in the ith position denoting $a_i = -1$ and `1` denoting $a_i = 1$.

If there are multiple answers, print any.

Example

standard input	standard output
2 1 5	1 1 2 11

Problem G. Function

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

Let's define the sum of all digits in x as $g(x)$. For example, $g(123) = 1 + 2 + 3 = 6$. Give you a function:

$$f(x) = Ax^2g(x) + Bx^2 + Cxg^2(x) + Dxg(x)$$

Find the minimum value of $f(x)$, where x is an integer and $1 \leq x \leq N$.

Input

This problem contains multiple test cases.

The first line of the input contains an integer $T (1 \leq T \leq 10^4)$, representing the number of test cases.

Each of the next T lines contains five integers $A, B, C, D, N (0 \leq |A| \leq 10^3, 0 \leq |B|, |C|, |D| \leq 10^6, 1 \leq N \leq 10^6)$ indicating a test case.

Output

For each test case output an integer, denoting the answer.

Example

standard input	standard output
2 1 2 3 4 100 4 3 2 1 100	10 10

Problem H. GCD on Sequence

Input file:	`standard input`
Output file:	`standard output`
Time limit:	5 seconds
Memory limit:	256 megabytes

You have a permutation a of length n. The value of an interval $[l, r]$ is defined as follow:

$$v(l, r) = \max_{l \le i < j \le r} \gcd(a_i, a_j)$$

In mathematics, the greatest common divisor(gcd) of two or more integers, which are not all zeroes, is the largest positive integer that divides each of the integers. For two integers x, y, the greatest common divisor of x and y is denoted by $\gcd(x, y)$. For example, the gcd of 8 and 12 is 4, that is, $\gcd(8, 12) = 4$.

You need to calculate for each $x (1 \le x \le n)$, how many pairs of $l, r (1 \le l < r \le n)$ satisfy that $v(l, r) = x$.

Input

This problem contains multiple test cases.

The first line contains an integer T indicating the number of test cases.

For each test case, the first line contains one integer n $(2 \le n \le 10^5)$.

The second line contains n integers a_i $(1 \le a_i \le n)$. It's guaranteed that a is a permutation.

It's guaranteed that $\sum n \le 10^6$.

Output

For each test case, output n lines.

The i-th line contains an integer indicating the answer for $x = i$.

Example

standard input	standard output
1 5 1 4 3 5 2	8 2 0 0 0

Problem I. Command Sequence

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

There is a robot that can move by receiving a sequence of commands.

There are four types of command in the command sequence:

- U: robot moves one unit up.
- D: robot moves one unit down.
- L: robot moves one unit left.
- R: robot moves one unit right.

Now, given a command sequence of length n. You need to find out how many substrings of the command sequence satisfy that if the robot execute the substring command, it can return to the starting position.

A substring is a contiguous sequence of characters within a string. For instance, `the best of` is a substring of `It was the best of times`. For example, `Itwastimes` is a subsequence of `It was the best of times`, but not a substring.

Input

This problem contains multiple test cases.

The first line contains an integer t ($1 \le t \le 20$) indicating the number of test cases.

For each test case, the first line contains one integer n ($2 \le n \le 10^5$).

The second line contains a $UDLR$ string of length n.

Output

For each test case, output one line one integer indicating the answer.

Example

standard input	standard output
1 6 URLLDR	2

Problem J. Random Walk

Input file:	`standard input`
Output file:	`standard output`
Time limit:	5 seconds
Memory limit:	256 megabytes

Give you an undirected simple graph with n vertices and m edges and there are a_i coins on the ith edge. Kris will randomly choose a starting vertex $s(1 \le s < n)$ and each second he will **uniformly** randomly walk to another adjacent vertex until reach the vertex n. The coins on edges will decay to a half per second but no less than b_i, formally if Kris walk through the ith edge at $t(t \ge 1)$ second, he will collect $max\{\lfloor \frac{a_i}{2^{t-1}} \rfloor, b_i\}$ coins. The graph will vary over time, after each change you should answer the expected coins he will collect. It is guaranteed that the graph is connected.

Input

This problem only contains one test case.

The first line of the input contains three integers $n, m, q (2 \le n \le 500, 1 \le m \le 10^4, 1 \le q \le 10^6)$. The next line contains $n-1$ integers $w_i (1 \le w_i \le 10^4)$. The probability of vertex x as the starting vertex can be represented as follow:

$$P_x = \frac{w_x}{\sum_{i=1}^{n-1} w_i} \tag{1}$$

The next m lines contains four integers $x, y, a, b (1 \le x, y \le n, 1 \le b \le a \le 100)$, representing an edge between x and y with a coins initially and b coins finally.

Then q lines follow, each line has one of the following format:

- 1 x y a b $(1 \le x, y \le n, 1 \le b \le a \le 100)$, representing changing the initial coins on (x, y) to a and final coins on (x, y) to b. It is guaranteed that there is an edge between x and y.
- 2 x c $(1 \le x \le n-1, 1 \le c \le 10^4)$, representing changing w_x to c. Note that after this change not only P_x but all probabilities will change according to formula (1). It is guaranteed that the number of this change is no more than n.

Note that the changes are persistent and the $(i+1)$th change is based on the ith change.

Output

Before the first change and after each change, output the expected coins Kris will collect. Assume that the answer is $\frac{P}{Q}$, then you should output $P \cdot Q^{-1} \mod 998244353$, where Q^{-1} denotes the multiplicative inverse of Q modulo 998244353.

Example

standard input	standard output
4 3 3 1 1 1 1 2 1 1 2 3 1 1 3 4 1 1 1 1 2 2 1 2 1 2 1 2 3 4 2	332748125 831870302 623902729 374341644

Note

- After Kris collect coins on an edge, the coins on it will **not** vanish. When he visit the edge the second time, he will collect coins on it again but the number may change.
- At the very begin, the expected coins which Kris will collect for starting vertices $1, 2, 3$ are 9 coins, 8 coins, 5 coins respectively, so the answer is $9 \times \frac{1}{3} + 8 \times \frac{1}{3} + 5 \times \frac{1}{3} = \frac{22}{3}$.

Problem K. Shooting Bricks

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1.5 seconds
Memory limit:	256 megabytes

Kris has a deep affection for a computer game named *Shooting Bricks*, described as follows:

The game is initialized with $n \times m$ bricks, placed in n rows and m columns. Players will be able to choose a column and shoot the outermost brick with a pistol. Shooting different bricks results in different bonus points, and shot bricks will disintegrate into ashes. Moreover, by shooting some special bricks, players can get a single bullet as a reward. It is the players' goal to achieve the highest overall bonus score.

Now Kris only has k bullets, and he wonders how many bonus points he can obtain at most.

Input

This problem contains multiple test cases.

The first line contains an integer $T(1 \leq T \leq 20)$, the number of test cases.

Each test case starts with a line of three integers $n, m, k(1 \leq n, m, k \leq 200)$.

Then n lines follow. The ith line contains m pairs of $f_{ij}, c_{ij}(1 \leq f_{ij} \leq 10^4, c_{ij} \in \{\mathtt{Y}, \mathtt{N}\})$, meaning if the brick located at the ith line and jth column is shot, player will get f_{ij} points and will(`Y`) or won't(`N`) get a reward bullet after that. Initally, the bricks at the nth line is exposed outside, so you need to shoot them as a start.

Output

For each test case, print a line of a single integer, denoting the highest bonus score.

Example

standard input	standard output
1 2 2 1 1 Y 1 N 1 Y 1 Y	4

Note

In the example, all bricks can be shot, shooting sequence $(2, 1), (1, 1), (2, 2), (1, 2)$ is one of the solutions.

Here is a link that may help you understand this problem. But please notice that it has nothing to do with this problem.

https://ccpc-online-2021.github.io/brick

Pay attention to the time you waste!

Problem L. Remove

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1.5 seconds
Memory limit:	256 megabytes

Kris is playing a game, which involves removing stones from a pile that initially consists of n stones. He is given a set of prime numbers P, and each time he is able to choose one p from P and remove $m \bmod p$ stones from the pile, where m stands for the number of stones remaining in the pile. The game finishes once no stone is left in the pile, and he wants to put an end to the game using least number of rounds possible.

Given N, you are expected to calculate the minimum rounds a_n required to finish the game starting with a pile of n stones for all $1 \le n \le N$. And if there is no possible solution for certain n, we set $a_n = 0$. And in order to prevent large output, you only need to print $\left(\sum_{1 \le n \le N} a_n \times 23333^{N-n}\right) \bmod 2^{64}$.

Input

This problem contains multiple test cases.

The first line contains an integer $T (1 \le T \le 60)$, the number of test cases.

Each case starts with a line of two integers $N, |P| (1 \le N \le 2 \times 10^6, 1 \le |P| \le 10^5)$.

Then $|P|$ prime integers follow, denoting the set. It is guaranteed that the primes are pairwise different and do not exceed N.

There will be no more than 15 test cases where $N \ge 20000$.

Output

Output T lines, each line contains an integer, the answer for the corresponding test case.

Example

standard input	standard output
1 6 2 2 3	17181031198765592570

Note

In the sample case, $\{a_i\}$ is $\{1, 1, 2, 3, 3, 0\}$.

Problem M. Start Dash ! !

Input file: `standard input`
Output file: `standard output`
Time limit: 4 seconds
Memory limit: 512 megabytes

It's time to start dash!!

As a LLer, Kris loves dashing and he wants everyone to enjoy the interest of dashing. Kris can dash towards someone and pick him up so that they will dash together, forming a ray. If Kris hits the wall after he picks someone up, they will both enjoy the interest of dashing. For some secret reason, the walls now form a convex hull and Kris can only start dashing form an arbitary point in a triangle outside the polygon. Now Kris wants you to calculate the area of points that he can enjoy dashing with.

Formally, there's a convex hull. You will be given m queries. Each query will give you a triangle A (**maybe degenerated**), which is strictly outside the polygon. You must answer the area of points which satisfy the condition:

1.The point is strictly outside the polygon.

2. There exists a point P in the triangle, the ray from P to this point intersects with the convex hull but the segment from P to this point does not.

Input

This problem contains multiple test cases.

The first line contains an intger T indicates the number of test cases.

For each test case, the frist line contains one intger n ($3 \le n \le 10^5$) indicating the number of points in the convex hull.

The next n lines each contains two integers x_i, y_i ($0 \le |x_i|, |y_i| \le 10^6$) which means the coordinate of the i_{th} point.

It's garanteed that the points will be given in the order to form the polygon and in counter-clockwise.

Then you will be given an integer q ($1 \le q \le 10^5$) indicating the number of queries.

The next q lines each contains six integers $x_1, y_1, x_2, y_2, x_3, y_3$ ($0 \le |x|, |y| \le 10^6$) which means the coordinates of the points of the triangle.

It's guaranteed that the sum of n is no more than 6×10^5.

Output

For each query, print the answer $2 \times S$ in one line. S indicates the area of points meets the condition.

It can be proved that $2 \times S$ is always an integer.

Example

standard input	standard output
1 8 -1 2 -2 1 -2 -1 -1 -2 1 -2 2 -1 2 1 1 2 1 0 3 0 4 1 5	11