Problem A. X-liked Counting

Input file:	`standard input`
Output file:	`standard output`
Time limit:	6 seconds
Memory limit:	128 megabytes

David is alway fond of new kinds of numbers.

He defines a number $x-liked$ if any of its prefixes or suffixes can be divided by x. A prefix is a contiguous part which begins at the leftmost position of the number and ends at any position of the number, and a suffix is a contiguous part which begins at any position of the number and ends at the rightmost position of the number. For example, 12 is a prefix of 123 while 23 is a suffix of it. The number itself can be a prefix or a suffix of itself.

Now David wants to know how many $x-liked$ integers there are in the range $[l, r]$. However, since David doesn't like the number 7, he won't allow digit 7 to occur in the numbers, so you don't need to count in numbers like 27,372,774 etc. because they all have at least one digit equals to 7.

Please tell him how many $x-liked$ integers there are in the range $[l, r]$ with no digit equals to 7.

Input

The first line contains one integer $T(1 \leq T \leq 10)$, the number of testcases.

For each testcase, there is one line which contains three non-negative integers, l,r and x.

$1 \leq l \leq r \leq 10^{18}$

$1 \leq x \leq 500$

Output

For each testcase, output one number in one line, how many $x-liked$ integers there are in the range $[l, r]$ with no digit equals to 7.

Example

standard input	standard output
1 11 20 4	5

Note

In the sample, there are five valid integers, 12 (12 can be divided by 4), 14 (4 can be divided by 4), 16 (16 can be divided by 4), 18 (8 can be divided by 4), 20 (20 and 0 both can be divided by 4).

Problem B. Buying Snacks

Input file:	`standard input`
Output file:	`standard output`
Time limit:	3 seconds
Memory limit:	256 megabytes

As a foodie, Diana loves eating snacks. One day, her snacks were all confiscated by Bella, so Diana decided to buy some more in a snack shop.

There are totally n different type of snacks in the snack shop, numbered from $1, 2, \ldots, n$. Each of snacks has two sizes of packages, *big* ones and *small* ones. Diana wants to taste different snacks as many as possible, so she will buy *at most* 1 *package* for each type of snacks. For any *two adjacent* kinds of snacks(no matter the size), Diana can choose to buy them as a bundle, so that she could have a discount compared to buying them respectively. To simplify this problem, we assume that any *small* package of snacks costs 1 coin, any *big* one costs 2 coins and any discount is 1 coin. Now Diana wonders how many different ways to buy snacks with *just* k coins.

Please output the answer module 998244353 for each $k \in [1, m]$.

Two ways are considered the *same* if types of snacks, sizes of snacks and bundles are all the same.

Notice: For *two adjacent* kinds of snacks, Diana could either buy them respectively without a discount or buy them as a bundle, and they are considered as different ways.

For example: when $n = 2, k = 2$, there are 5 different ways:

$(1b), (2b), (1s\&2b), (1b\&2s), (1s)(2s)$

(number means the type of snack, s means its a small package, b means its a big package, & means its a bundle)

Input

The first line contains an integer $T (T \le 5)$, denoting the number of test cases. Each test case contains two integers $n (1 \le n \le 10^9)$, $m (1 \le m \le 20000)$ denoting the number of types of snacks and the maximum number of coins. It is guaranteed that for all test cases, $\sum m \le 30000$

Output

For each test case, output a line with m integers, indicating the answer.

Example

standard input	standard output
2 2 3 5 4	3 5 3 9 38 97 166

Problem C. Ink on paper

Input file:	`standard input`
Output file:	`standard output`
Time limit:	5 seconds
Memory limit:	512 megabytes

Bob accidentally spilled some drops of ink on the paper. The initial position of the i-th drop of ink is (x_i, y_i), which expands outward by 0.5 centimeter per second, showing a circle. The curious Bob wants to know how long it will take for all the inks to become connected. In order to facilitate the output, please output the square of the time.

Input

The first line of input contains one integer $T(1 \leq T \leq 5)$, indicating the number of test cases. For each test case, the first line contains one integer $n(2 \leq n \leq 5000)$, indicating the number of ink on the paper. Each of the next n lines contains 2 integers $(x_i, y_i)(|x_i| \leq 10^9, |y_i| \leq 10^9)$, indicating that x and y coordinates of the ink.

Output

For each test case, output one line containing one decimal, denoting the answer.

Example

standard input	standard output
2 3 0 0 1 1 0 1 5 1 1 4 5 1 4 2 6 3 10	1 17

Problem D. Counting Stars

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

As an elegant and accomplished girl, Bella loves watching stars at night (especially the Polaris). There are totally n stars in the sky. We can define a_i as the initial *brightness* of i-th star.

At t-th second, the *brightness* of some stars may change due to stars' activity. To simplify this problem, we can approximately divide activities into two kinds:

1. $\forall i \in [l, r]$, the *brightness* of i-th star decreases by $a_i \& (-a_i)$

2. $\forall i \in [l, r], a_i \neq 0$, the *brightness* of i-th star increases by 2^k, where k satisfy $2^k \leq a_i < 2^{k+1}$

Also, Bella has some queries. For each query, she wants to know the *sum* of *brightness* of stars in the range $[l, r]$, which means $\sum_{i=l}^{r} a_i$

As we all know, Bella is a big smart. Thus, she could not solve such difficult problem by herself. Please help her solve this problem and answer her queries.

Since the answer may be too large, please output the answer modulo 998244353

Input

The first line contains an integer $T (T \leq 5)$, denoting the number of test cases.

For each test case:

The first line contains an integers $n (1 \leq n \leq 100000)$, denoting the number of stars .

The second line contains n integers $a_1, a_2, \ldots, a_n (1 \leq a_i \leq 10^9)$, denoting the initial *brightness* of stars.

The third line contains an integer $q (1 \leq q \leq 100000)$, denoting the number of operations.

Each of the next q lines contains three integers opt_i, l_i, r_i, denoting the kind of operations and ranges.

$opt_i = 1$ means Bella's query.

$opt_i = 2$ means the first kind of stars' activity.

$opt_i = 3$ means the second kind of stars' activity.

It is guaranteed that for all test cases, $\sum n \leq 4 \times 10^5, \sum q \leq 4 \times 10^5$.

Output

For each Bella's query, output the answer modulo 998244353 in a single line.

Example

standard input	standard output
1 5 5 2 2 9 7 4 2 1 5 1 1 1 3 1 3 1 2 5	4 14

Problem E. Separated Number

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

Cathy loves numbers, and recently she fell in love with the separation of numbers.

A separation of a number is defined as dividing the number into contiguous parts. For example, we can call (11)(451)(4) a separation of the number 114514. The value of one separation is the sum of all the separated parts(the value of (11)(451)(4) equals to 11+451+4=466). If one part has leading zeros, it is also valid, so the separation (1)(00) of number 100 is a valid separation too. Now Cathy has a number x without leading zeros, and she wants to know the total value of separations which divide the number into no more than k parts. She is not quite smart so she asked you for help.

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

Input

The first line contains a number $T(1 \leq T \leq 5)$, the number of testcases.

For each testcase, there are two lines.

The first line contains a number k, the maximum number of parts.

The second line contains a number x, the queried number.

Let n be the number of digits of x, and we will have $1 \leq n \leq 10^6$ and $1 \leq k \leq n$.

It is guaranteed that for all testcases, $\sum n \leq 10^6$.

Output

For each testcase, output one number in one line, the answer modulo 998244353.

Example

standard input	standard output
1 3 100	112

Note

In the sample, there are 4 possible separations with no more than 3 parts, (100) , (1)(00) , (10)(0) , (1)(0)(0) , and their values are 100 , $1 + 0 = 1$, $10 + 0 = 10$, $1 + 0 + 0 = 1$ respectively, so the answer will be $100 + 1 + 10 + 1 = 112$.

Problem F. GCD Game

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

Alice and Bob are playing a game.

They take turns to operate. There are n numbers, a_1 , a_2 , ... , a_n. Every time, the player plays in 3 steps.

1.Arbitrarily chooses one number a_i.

2.Arbitrarily chooses another number $x(1 \leq x < a_i)$.

3.Replace the number a_i with $gcd(a_i, x)$. Here, $gcd(u, v)$ refers to the **Greatest Common Divisor** of u and v.

When a player can not make a single move he/she loses the game. Alice moves the first and she asks you to tell her who will win the game if both player play optimally.

Input

The first line contains a number $T(1 \leq T \leq 100)$, the number of testcases.

For each testcase, there are two lines.

The first line contains one number $n(1 \leq n \leq 10^6)$.

The second line contains n numbers a_1 , a_2 , ... , $a_n(1 \leq a_i \leq 10^7)$.

Output

For each testcase, output the answer in one line.

If Alice will win the game, print **Alice** , otherwise print **Bob** .

Example

standard input	standard output
2 1 1 1 2	Bob Alice

Problem G. A Simple Problem

Input file:	`standard input`
Output file:	`standard output`
Time limit:	2 seconds
Memory limit:	512 megabytes

You have a sequence A of length n and a positive integer k. Initially, all elements in A are set to 0.

Now there are q operations, these operations can be divided into two types.

$1\ l\ r\ x : \forall i \in [l, r] A_i = A_i + x$

$2\ l\ r$: Find $\min\limits_{i=l}^{r-k+1} (\max\limits_{j=i}^{i+k-1} a_j)$

Input

The first line contains an integer $T (T \le 5)$, denoting the number of test cases. Each test case contains $q + 2$ lines

The first line contains three integer $n, k (2 \le k \le n \le 5 \times 10^8)$ and $q (1 \le q \le 1 \times 10^5)$.

The next q lines describe operations of two types:

$1\ l\ r\ x : \forall i \in [l, r] A_i = A_i + x$

$2\ l\ r$: Find $\min\limits_{i=l}^{r-k+1} (\max\limits_{j=i}^{i+k-1} a_j)$

It is guaranteed that the sum of q won't exceed 2×10^5.

Output

For each operation of type 2, output the answer in a single line.

Example

standard input	standard output
2 5 3 3 1 2 5 2 1 3 4 -1 2 1 4 10 4 10 1 1 6 6 1 3 8 -6 2 2 6 1 4 8 -8 1 4 9 4 1 4 5 -7 2 4 8 1 6 7 8 1 1 3 -2 2 3 7	2 0 -4 4

Problem H. Square Card

Input file: `standard input`
Output file: `standard output`
Time limit: 1 second
Memory limit: 64 megabytes

Eric is playing a game on an infinite plane.

On the plane, there is a circular area with radius r_1 called **the Scoring Area**. Every time Eric will throw a square card with side length a into the plane, then the card will start rotating around its centre. If at one moment the card is strictly inside **the Scoring Area**, the current play will be scored.

There is another circular area with radius r_2 called **the Bonus Area**. If the play is scored in the current situation, the square card will continue rotating and if at one moment it is strictly inside **the Bonus Area**, he will get an extra bonus.

Eric isn't good at this game, so we can briefly consider that he will throw the card to any postion with an equal probability.

Now Eric wants to know what the ratio of the possibility of being scored and getting the bonus simultaneously to the possibility of being scored is.

The coordinates of the centre of **the Scoring Area** are (x_1, y_1).

The coordinates of the centre of **the Bonus Area** are (x_2, y_2).

Input

The first line contains a number $T(1 \le T \le 20)$, the number of testcases.

For each testcase, there are three lines.

In the first line there are three integers r_1,x_1,$y_1$$(-1000 \le x_1, y_1 \le 1000, 0 < r_1 \le 1000)$, the radius and the postion of **the Scoring Area**.

In the second line there are three integers r_2,x_2,$y_2$$(-1000 \le x_2, y_2 \le 1000, 0 < r_2 \le 1000)$, the radius and the postion of **the Bonus Area**.

In the third line there is an integer $a(0 < a \le 1000)$, the side length of the square card.

It is guaranteed that the Scoring Area is big enough thus there is always a chance for the play to be

Output

For each test case, output a decimal $p(0 \le p \le 1)$ in one line, the ratio of the possibility of being scored and getting the bonus simultaneously to the possibility of being scored.

Your answer should be rounded to 6 digits after the decimal point.

Example

standard input	standard output
1 5 1 2 3 2 1 1	0.301720

Problem I. Singing Superstar

Input file:	`standard input`
Output file:	`standard output`
Time limit:	2 seconds
Memory limit:	512 megabytes

Ke Ke is a cheerful, enthusiastic girl. She dreams of singing happily every day. One day, while practicing singing, Ke Ke accidentally discovered that there are some words that can make the song better. She called these words "$SuperstarWords$".

After a lot of attempts, she found that in each song, there are a total of n "$SuperstarWords$". She wants to know the $maximum$ number of $disjoint$ occurrences of each "$SuperstarWords$"in her song.

$Note$: one occurrence means that the $SuperstarWord$ is a continuous substring of the song at a certain position.

For example: The $maximum$ number of $disjoint$ occurrences of "aba"in "abababa"is 2

Input

The first line contains an integer $T(T \leq 5)$, denoting the number of test cases.

The first line of each test case contains one string $s(1 \leq |s| \leq 10^5)$, denoting a song.

The second line of each test case contains one integer $n(1 \leq n \leq 10^5)$, denoting the number of $SuperstarWords$

The next n lines contains contains one string $a_i(1 \leq |a_i| \leq 30)$, denoting one $SuperstarWords$

It is guaranteed that for all test cases, $\sum |s| \leq 4 * 10^5$, $\sum |a_i| \leq 4 * 10^5$

$Note$:All input string character sets are lowercase English letters

Output

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

Example

standard input	standard output
2 abababbbaanbananananabanana 3 aba ababa nana nihaoxiexiexiaolongbaozaijian 3 qwer taihaotinleba xiexi	2 1 2 0 0 1

Problem J. Yinyang

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

You have a grid of n rows and m columns, each cell should be painted either black or white.

The cell in the ith row and jth column is denoted as (i, j).

Two cells are *directly connected* if and only if they have a common edge.

Two cells are *connected* if and only if they are *directly connected* or there exist a cell *connected* to both cells.

A painting plan is *good* if and only if it satisfy three conditions:

1. All white cells are *connected*

2. All black cells are *connected*

3. $\forall 1 \le i < n, 1 \le j < m$, cell$(i, j)$,$(i, j + 1)$,$(i + 1, j)and(i + 1, j + 1)$ can't have the same color.

Some of the cells have been painted, you should paint the rest .

output the number of *good* painting plans module 998244353.

Input

The first line of input contains an integer $T (T \le 10)$, denoting the number of test cases.

Each test case contains $n + 1$ lines.

The first line contains two integer $n, m(3 \le n \le 100, 3 \le m \le 100, nm <= 100)$, denoting the size of the grid.

The next n lines describe the painted cells, each line contains m integer.

The jth number in ith row describe cell(i, j),the number is 0,1 or -1.

0 means the cell is painted white,1 means the cell is painted black, -1 means the cell is not painted.

Output

Output the number of *good* painting plans module 998244353.

Example

standard input	standard output
3 3 3 1 0 0 1 1 0 -1 -1 0 3 4 1 -1 -1 0 -1 -1 -1 -1 0 -1 -1 1 10 10 -1	2 0 139719073