Problem A. 01 Matrix

Input file:	`standard input`
Output file:	`standard output`
Time limit:	2 seconds
Memory limit:	1024 megabytes

There are 2^{NM} matrices $A = (A_{i,j})\ (1 \le i \le N,\ 1 \le j \le M)$ of size N rows and M columns consisting only of 0 and 1. Among them, find the number of matrices that satisfy the following condition, and output the result modulo 998244353.

- For all $k = 1, 2, \ldots, K$, both of the following hold:
  - $\displaystyle\sum_{i=1}^{x_k}\sum_{j=1}^{y_k} A_{i,j}$ is odd.
  - $\displaystyle\sum_{i=x_k+1}^{N}\sum_{j=y_k+1}^{M} A_{i,j}$ is odd.

Input

The input is given in the following format:

N M
K
x_1 y_1
x_2 y_2
$\vdots$
x_K y_K

- $2 \le N, M \le 10^9$
- $1 \le K \le 3 \times 10^5$
- $1 \le x_i < N\ (1 \le i \le K)$
- $1 \le y_i < M\ (1 \le i \le K)$
- $(x_i, y_i) \ne (x_j, y_j)\ (i \ne j)$
- All input values are integers.

Output

Print the answer.

Examples

standard input	standard output
3 4 2 2 2 1 3	256
76 38 4 7 6 3 8 20 26 3 28	361562686

Note

For the first sample input, the matrix above the figure satisfies the condition. However, the matrix below does not satisfy the condition because $\sum_{i=1}^{x_1}\sum_{j=1}^{y_1} A_{i,j}$ is even.

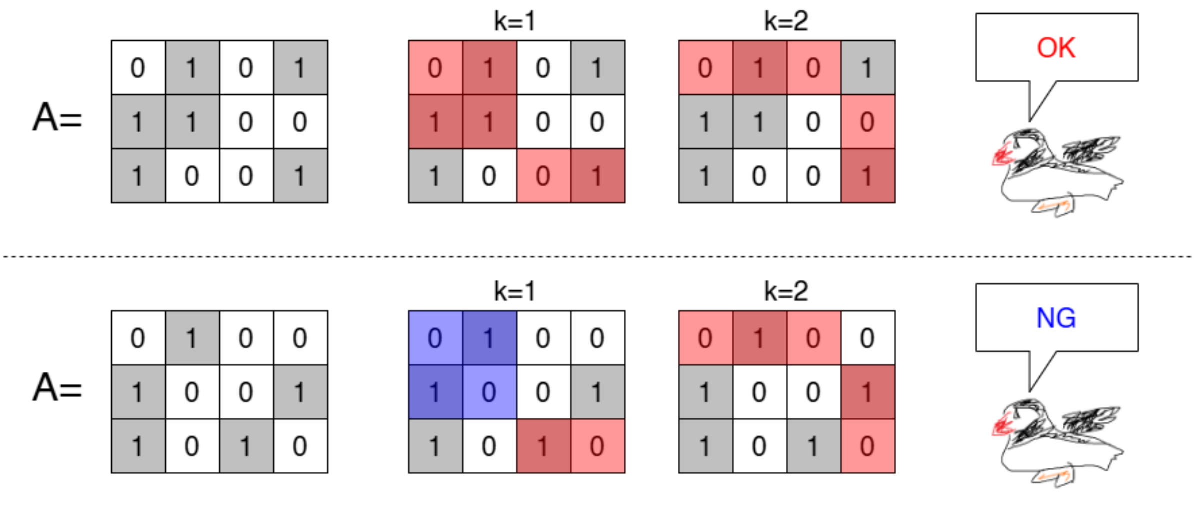

Рис. 1: Explanation for the first example.

For the second example, do not forget to output the answer modulo 998244353.

Problem B. Birds-of-Paradise' Card Game

Input file:	`standard input`
Output file:	`standard output`
Time limit:	3 seconds
Memory limit:	1024 megabytes

You are given a pair of non-negative integers W, S.

Puffin Pataro is playing a card game by himself. There are two types of cards: card w and card s.

Initially, Pataro has W cards of type w and S cards of type s, and his score is 0. He repeatedly consumes one card from the cards he has until all cards are consumed. When the i-th consumed card is a card s, the following effect occurs.

- Among the $(i-1)$-th, $(i-2)$-th, and $(i-3)$-th consumed cards, let x be the number of cards that are card w. He gains a score of $27 \times \left(\frac{4}{3}\right)^x$.

For convenience, assume that cards consumed at positions 0 or earlier are not card w.

Find the maximum possible total score that Pataro can obtain by consuming the cards in an appropriate order. It can be proven that the required answer is always an integer.

Solve the above problem for T test cases.

Input

The input is given in the following format:

T
case_1
case_2
$\vdots$
case_T

Each test case is given in the following format:

W S

- $1 \le T \le 10^4$
- $0 \le W, S \le 10^{16}$
- All input values are integers.

Output

Print T lines.

The i-th line should contain the answer for the i-th test case.

Example

standard input	standard output
3 5 3 12 9 2026 328	160 484 20992

Note

In the first example, it is optimal for Pataro to consume the cards in the order w, w, s, s, w, w, w, s.

- The 3-rd consumed card is a card s. Among the 2-nd, 1-st, and 0-th consumed cards, there are 2 cards of type w, so he gains a score of $27 \times \left(\frac{4}{3}\right)^2 = 48$.
- The 4-th consumed card is a card s. Among the 3-rd, 2-nd, and 1-st consumed cards, there are 2 cards of type w, so he gains a score of $27 \times \left(\frac{4}{3}\right)^2 = 48$.
- The 8-th consumed card is a card s. Among the 7-th, 6-th, and 5-th consumed cards, there are 3 cards of type w, so he gains a score of $27 \times \left(\frac{4}{3}\right)^3 = 64$.

Therefore, the total score Pataro can obtain is 160. No matter how the cards are consumed, it is not possible to obtain a score greater than 160, so the answer is 160.

Problem C. Don't be Clockwise

Input file:	`standard input`
Output file:	`standard output`
Time limit:	2 seconds
Memory limit:	1024 megabytes

There are N points $p_1, p_2, \ldots, p_N$ on a two-dimensional plane.

Point p_i is located at (X_i, Y_i). No two points share the same coordinates.

Construct one permutation $q = (q_1, q_2, \ldots, q_N)$ of the sequence of points $p = (p_1, p_2, \ldots, p_N)$ that satisfies the following condition, or determine that it is impossible.

- For all $1 \le i \le N-2$, the three points q_i, q_{i+1}, q_{i+2} are either collinear or form a counterclockwise turn in this order.
  - More precisely, let the coordinates of q_i be (x_i, y_i). Then the following must hold: $(x_{i+1} - x_i)(y_{i+2} - y_{i+1}) - (y_{i+1} - y_i)(x_{i+2} - x_{i+1}) \ge 0$.

Solve this problem for T test cases.

Input

The input is given in the following format:

T
case_1
case_2
$\vdots$
case_T

Each test case is given in the following format:

N
X_1 Y_1
X_2 Y_2
$\vdots$
X_N Y_N

- $1 \le T \le 100$
- $3 \le N \le 3000$
- $0 \le X_i, Y_i \le 10^9$
- $(X_i, Y_i) \ne (X_j, Y_j)$ $(i \ne j)$
- The sum of N over all test cases does not exceed 3000
- All input values are integers

Output

Print T lines.

For the i-th line, if there is no q satisfying the condition for the i-th test case, print -1.

Otherwise, let $q = (p_{r_1}, p_{r_2}, \ldots, p_{r_N})$. Output $r_1, r_2, \ldots, r_N$ in this order, separated by spaces.

If multiple valid answers exist, you may output any of them.

Example

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

Note

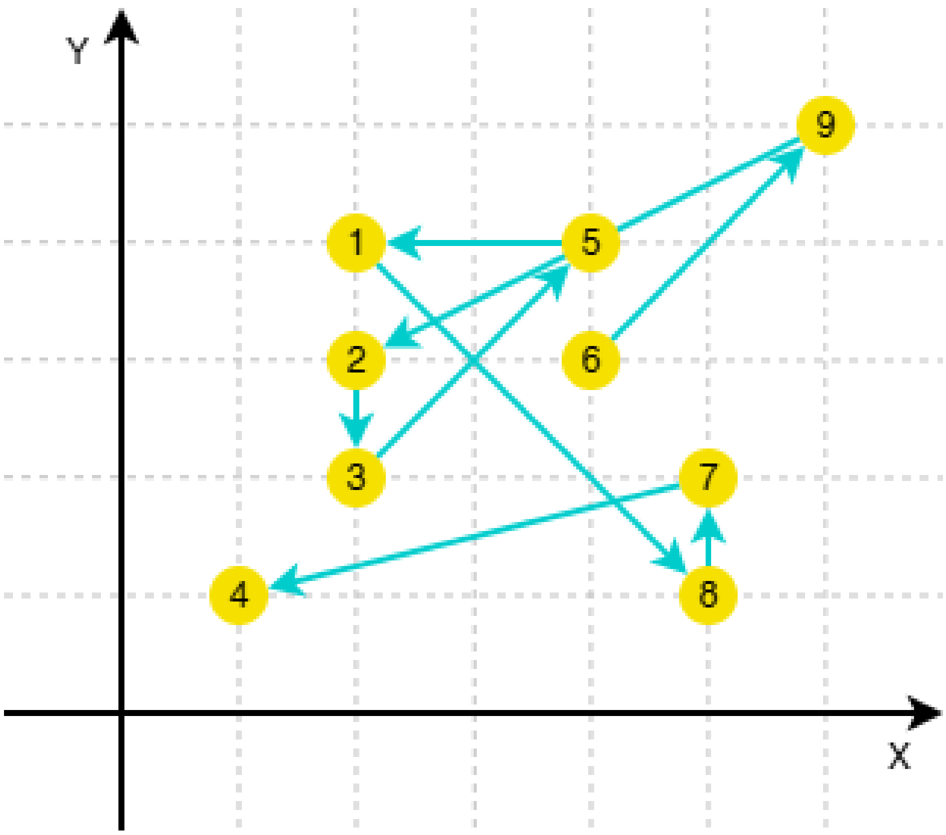

Рис. 2: Explanation for the example.

Problem D. Except Ai

Input file: `standard input`
Output file: `standard output`
Time limit: 2 seconds
Memory limit: 1024 megabytes

You are given integers N and X, and an integer sequence $A = (A_1, A_2, \ldots, A_N)$ of length N. You will construct an integer sequence $s = (s_1, s_2, \ldots, s_N)$ of length N that satisfies all of the following conditions:

- $1 \le s_i \le X$
- $s_i \ne A_i$ $(1 \le i \le N)$

Find the maximum possible number of indices i $(1 \le i < N)$ such that $s_i = s_{i+1}$.

Input

The input is given in the following format:

N X
A_1 A_2 $\ldots$ A_N

- $1 \le N \le 5 \times 10^5$
- $2 \le X \le 10^9$
- $1 \le A_i \le X$ $(1 \le i \le N)$
- All input values are integers.

Output

Print the answer.

Examples

standard input	standard output
8 3 1 2 2 3 2 1 2 1	5
4 8 7 6 3 8	3

Note

In the first test case, for example, if we choose $s = (3, 3, 1, 1, 3, 3, 3, 3)$, then the indices i such that $s_i = s_{i+1}$ are $1, 3, 5, 6, 7$, for a total of 5. It can be shown that it is impossible to make $s_i = s_{i+1}$ hold for more indices than this, so the answer is 5.

In the second test case, for example, we can choose $s = (5, 5, 5, 5)$.

Problem E. Find "rururutata"

Input file:	`standard input`
Output file:	`standard output`
Time limit:	3.5 seconds
Memory limit:	1024 megabytes

A **rururutata type sequence** is defined as a sequence that can be written as $r + r + r + t + t$ (+ means concatenation) using **non-empty** sequences r and t.

You are given an integer sequence $S = (S_1, S_2, \ldots, S_N)$ of length N. There are Q queries, which you must process in order.

In each query, integers L and R are given. Determine whether there exists a **rururutata type sequence** as a contiguous subarray of $(S_L, S_{L+1}, \ldots, S_R)$.

Input

The input is given in the following format:

N
$S_1\ S_2\ \ldots\ S_N$
Q
query_1
query_2
$\vdots$
query_Q

Each query is given in the following format:

$L\ R$

- $1 \le N \le 5 \times 10^5$
- $1 \le S_i \le N\ (1 \le i \le N)$
- $1 \le Q \le 5 \times 10^5$
- $1 \le L \le R \le N$
- All input values are integers.

Output

Print Q lines.

On the i-th line, print `Yes` if a **rururutata type sequence** exists for the i-th query, and `No` otherwise.

Example

standard input	standard output
17 3 3 3 2 2 2 4 3 4 3 4 3 2 2 2 2 2 5 1 5 4 12 2 6 8 15 13 17	Yes Yes No No Yes

Note

For the first query, $(S_1, S_2, S_3, S_4, S_5) = (3, 3, 3, 2, 2)$ is a **rururutata type sequence**. The condition is satisfied by taking $r = (3)$ and $t = (2)$.

For the second query, among $(S_4, S_5, \ldots, S_{12})$, the contiguous subarray $(S_4, S_5, \ldots, S_{10}) = (2, 2, 2, 4, 3, 4, 3)$ is a **rururutata type sequence**. The condition is satisfied by taking $r = (2)$ and $t = (4, 3)$.

For the third query, $(3, 3, 2, 2, 2)$ is not a **rururutata type sequence**, and it clearly does not contain any shorter **rururutata type sequence** either.

For the fourth query, note that $(3, 3, 3, 2, 2)$ is not the contiguous subarray.

For the fifth query, note that it is allowed for r and t to be the same.

Problem F. Increase Decrease

Input file:	`standard input`
Output file:	`standard output`
Time limit:	2 seconds
Memory limit:	1024 megabytes

You are given two sequences A and B of length N.

For all i ($1 \le i \le N$), determine whether there exists a permutation C of $(1, 2, \ldots, N)$ such that the LIS (Longest Increasing Subsequence) of the prefix of length i is A_i, and the LDS (Longest Decreasing Subsequence) of the prefix of length i is B_i.

If such a permutation exists, construct one.

Input

The input is given in the following format:

N
$A_1\ A_2\ \ldots\ A_N$
$B_1\ B_2\ \ldots\ B_N$

- $1 \le N \le 5 \times 10^5$
- $1 \le A_i \le N$ $(1 \le i \le N)$
- $1 \le B_i \le N$ $(1 \le i \le N)$
- All input values are integers

Output

If no such permutation exists, print -1. Otherwise, print N space-separated integers representing the elements of the permutation C. If there are multiple such permutations, any of them will be accepted.

Examples

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

Problem G. Make T

Input file:	`standard input`
Output file:	`standard output`
Time limit:	3 seconds
Memory limit:	1024 megabytes

There are N^2 aliens standing at every integer lattice point (x, y) such that $0 \le x, y < N$. No aliens exist at any other locations. Each alien has four hands. Two aliens can hold hands with each other (a bidirectional connection) if the distance between them is exactly 1.

Initially, some pairs of aliens are already holding hands as follows:

- If $A_{i,j} = 1$, the alien at (i, j) and the alien at $(i, j + 1)$ are already holding hands $(0 \le i < N, 0 \le j < N - 1)$.
- If $B_{i,j} = 1$, the alien at (i, j) and the alien at $(i + 1, j)$ are already holding hands $(0 \le i < N - 1, 0 \le j < N)$.

You can perform the following operation zero or more times:

- Choose two aliens who are at a distance of exactly 1 from each other and have them hold hands.

By performing these operations optimally, maximize the number of aliens who are holding hands with exactly three other aliens.

Input

The input is given in the following format:

N
$A_{0,0}A_{0,1} \ldots A_{0,N-2}$
$\vdots$
$A_{N-1,0}A_{N-1,1} \ldots A_{N-1,N-2}$
$B_{0,0}B_{0,1} \ldots B_{0,N-1}$
$\vdots$
$B_{N-2,0}B_{N-2,1} \ldots B_{N-2,N-1}$

- $2 \le N \le 500$
- $A_{i,j} \in \{0, 1\}$ $(0 \le i < N, 0 \le j < N - 1)$
- $B_{i,j} \in \{0, 1\}$ $(0 \le i < N - 1, 0 \le j < N)$
- All input values are integers.

Output

Print the maximum possible number of aliens who are holding hands with exactly three other aliens.

Examples

standard input	standard output
4 000 010 010 000 0000 0110 0000	8
7 001110 111011 100110 101111 100001 001001 010110 0101011 1001110 1101000 0001111 0111000 0110111	42

Note

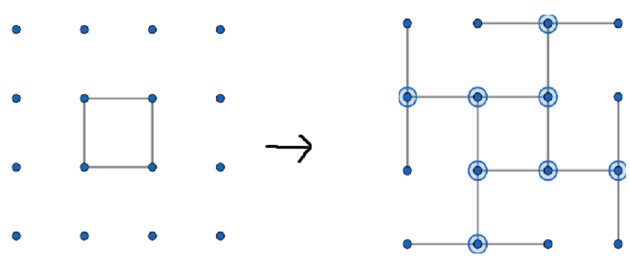

Рис. 3: Initial configuration and an optimal way to add hand connections for the sample.

Sample Explanation The initial configuration of the sample is shown in the left part of Figure 1. As illustrated in the figure, by additionally connecting aliens' hands **12 times**, we can make **8 aliens** each hold hands with **exactly three other aliens** in total. This is the maximum possible number, and therefore the result is optimal.

Problem H. OR Preference

Input file:	`standard input`
Output file:	`standard output`
Time limit:	3 seconds
Memory limit:	768 megabytes

Note that the memory constraints are special.

You are given an integer N and an integer sequence $A = (A_1, A_2, \ldots, A_N)$.

You repeatedly perform operations on the sequence until its length becomes 1. At each step, you choose two adjacent elements in the current sequence and merge them into a single element. After each merge, the resulting sequence is reindexed in the natural order. The following two types of operations are available.

- Operation AND: replace A_i and A_{i+1} with $(A_i \& A_{i+1})$.
- Operation OR: replace A_i and A_{i+1} with $(A_i | A_{i+1})$.

Here, the operators & and | denote the bitwise AND and bitwise OR operations, respectively.

Since each operation reduces the length of the sequence by 1, exactly $(N-1)$ operations are performed in total.

Among all possible sequences of operations such that the final remaining element is equal to 0, find the maximum possible number of times Operation OR is performed.

Input

The input consists of multiple test cases.

```
T
testcase 1
testcase 2
⋮
testcase T
```

Each test case is given in the following format:

```
N
A_1, A_2, ..., A_N
```

- $1 \leq T \leq 2^{12}$
- $2 \leq N \leq 2^{13}$
- $0 \leq A_i < 2^{13}$
- The sum of N over all test cases does not exceed 2^{13}.
- All input values are integers.

Output

Print the maximum number of times Operation OR can be performed among all operation sequences that satisfy the condition that the final remaining element is equal to 0. If no sequence of operations satisfies the condition, print -1.

Example

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

Note

This input contains 4 test cases.

For the first test case, we can perform the second type of operation 3 times as follows:

- Initially, $A = (3, 0, 1, 4, 1, 5)$.
- Apply the OR operation to (A_3, A_4). After the operation, $A = (3, 0, 5, 1, 5)$.
- Apply the AND operation to (A_1, A_2). After the operation, $A = (0, 5, 1, 5)$.
- Apply the OR operation to (A_3, A_4). After the operation, $A = (0, 5, 5)$.
- Apply the OR operation to (A_2, A_3). After the operation, $A = (0, 5)$.
- Apply the AND operation to (A_1, A_2). After the operation, $A = (0)$.

Since the final state satisfies $A_1 = 0$, the condition is met.

It is impossible to satisfy the condition with 4 or more OR operations, so the answer is 3.

Problem I. Penguin Flicker

Input file: `standard input`
Output file: `standard output`
Time limit: 3.5 seconds
Memory limit: 1024 megabytes

There is a long horizontal skating rink. The rink is divided into $L+2$ sections, numbered from left to right as $0, 1, 2, \ldots, L, L+1$. Sections 0 and $L+1$ contain holes that lead into the sea, while the other sections have no holes.

Among the sections without holes, N sections contain penguins. The i-th penguin is located in section P_i, and all sections containing penguins are distinct.

Puffin Pataro will now move the penguins until all of them have fallen into the sea. Specifically, the following operation is repeated until all penguins have fallen into the sea.

- Uniformly at random, choose one penguin that has not yet fallen into the sea.
- Uniformly at random, choose either left or right, and move the chosen penguin in that direction. The penguin continues moving in the chosen direction until one of the following conditions is satisfied:
  - It reaches the section immediately before a section that contains another penguin.
  - It reaches a section with a hole and falls into the sea.

A penguin that has fallen into the sea is considered to be in no section. All random choices are made independently.

When a penguin moves from section i to section j in a single operation, the distance moved in that operation is defined as $|i-j|$. Compute the expected value of the total distance moved by all penguins until all of them have fallen into the sea, modulo 998244353.

Solve this problem for T test cases.

Input

The input is given in the following format:

```
T
case_1
case_2
⋮
case_T
```

Each test case is given in the following format:

```
N L
P_1 P_2 ... P_N
```

- $1 \le T \le 100$
- $1 \le N \le 5000$
- $N \le L \le 10^9$
- $1 \le P_1 < P_2 < \cdots < P_N \le L$
- All inputs are integers.

Output

Print T lines.

On the i-th line, output the answer for the i-th test case. More precisely, it can be shown that the expected value is always a rational number. Under the constraints of this problem, when the value is written as $\frac{p}{q}$ using coprime positive integers p and q, there exists a unique integer r such that $r \times q \equiv p \pmod{998244353}$ and $0 \le r < 998244353$. Output this integer r.

Example

standard input	standard output
3 1 8 2 4 7638 7 66 333 888 5 21 2 4 9 15 17	499122181 308996191 485077673

Note

In the first test case, if Pataro moves the first penguin to the left, the distance moved is 2, and if he moves it to the right, the distance moved is 7. In either case, the penguin falls into the sea. Therefore, the expected distance moved is $\frac{9}{2}$. Since $499122181 \times 2 \equiv 9 \pmod{998244353}$, the answer is 499122181.

In the second testcase, for instance, if Pataro first moves the fourth penguin to the left, the fourth penguin moves to section 334, and the distance moved in this operation is 554. There are multiple possible ways to move the penguins, and Pataro chooses one of them uniformly at random.

Problem J. Set Sequence

Input file:	`standard input`
Output file:	`standard output`
Time limit:	4 seconds
Memory limit:	1024 megabytes

You are given a positive integer N, a prime number P, and a sequence of positive integers $A = (A_1, A_2, \ldots, A_N)$ of length N, where each element is less than P. Let $S = \{1, 2, \ldots, N\}$.

Find the number of sequences T of sets with length at least 1 that satisfy all of the following conditions, and output the result modulo P.

- Every element of T is a **non-empty** subset of S.
- For each $i = 1, 2, \ldots, N$, the following holds:
  - Among the elements of T, exactly A_i of them contain i.

Input

The input is given in the following format:

```
N P
A_1 A_2 ... A_N
```

- $1 \le N \le 2 \times 10^5$
- $113 \le P \le 500009$
- P is a prime number
- $1 \le A_i < P$ $(1 \le i \le N)$
- All input values are integers

Output

Print the answer.

Examples

standard input	standard output
2 113 1 2	5
4 8191 7 6 3 8	4477

Note

For the first example, the possible sequences T are $(\{1,2\},\{2\})$, $(\{2\},\{1,2\})$, $(\{1\},\{2\},\{2\})$, $(\{2\},\{1\},\{2\})$, $(\{2\},\{2\},\{1\})$, for a total of 5.

For the second example, do not forget to find the number of valid sequences T modulo P.

Problem K. Talk Event

Input file:	`standard input`
Output file:	`standard output`
Time limit:	2 seconds
Memory limit:	1024 megabytes

N people have applied for a voice actor talk event. To apply for this event, a person must purchase between 1 and 4 tickets inclusive, and there are T_i people who purchased i tickets ($1 \le i \le 4$). **Here, people are distinguished only by the number of tickets they purchased; people with the same number of tickets are considered indistinguishable.**

In this talk event, if a person wins the lottery, they can talk with the voice actor for a duration equal to the number of tickets they purchased.

The total duration of the event is X units of time. Puffin Pataro, the god of randomness, selects the winners so that the total duration does not exceed the event time (in other words, the sum of the numbers of tickets purchased by the winners is at most X). However, to avoid complaints from those who are not selected, the following condition must be satisfied:

- For any person who is not selected, if that person were additionally selected, the total duration would exceed the event time.

Find the number of ways to select **exactly** K winners, modulo 998244353.

Solve this problem for TESTCASES test cases.

Input

The input is given in the following format:

```
TESTCASES
case_1
case_2
⋮
case_T
```

Each test case is given in the following format:

```
N K X
T_1 T_2 T_3 T_4
```

- $1 \le \text{TESTCASES} \le 10^4$
- $1 \le K \le N \le 2.5 \times 10^8$
- $1 \le X \le 10^9$
- $0 \le T_i$ $(1 \le i \le 4)$
- $T_1 + T_2 + T_3 + T_4 = N$
- All input values are integers

Output

Output TESTCASES lines.

On the i-th line, output the answer for the i-th test case.

Example

standard input	standard output
2 24 8 11 7 6 3 8 24 3 28 7 6 3 8	4 0

Note

Let t_i be the number of winners among the people who purchased i tickets. In the first test case, there are the following 4 possible selections:

- $(t_1, t_2, t_3, t_4) = (5, 3, 0, 0)$
- $(t_1, t_2, t_3, t_4) = (6, 1, 1, 0)$
- $(t_1, t_2, t_3, t_4) = (7, 0, 0, 1)$
- $(t_1, t_2, t_3, t_4) = (7, 0, 1, 0)$

In the second test case, there is no selection that satisfies the conditions.

Problem L. Unique Sheet

Input file:	`standard input`
Output file:	`standard output`
Time limit:	3 seconds
Memory limit:	1024 megabytes

There is an $N \times N$ grid where each cell (i, j) contains an integer $A_{i,j}$ $(1 \le A_{i,j} \le (N-K)^2)$. Pataro the Panda performed the following operations on this grid:

- Choose K rows out of the N rows and delete them.
- Choose K columns out of the N columns and delete them.

After performing these operations, it is said that all remaining $(N-K)^2$ integers in the grid are distinct. Find the number of ways Pataro could have performed these operations, modulo 998244353.

Two operations are considered different if and only if the set of chosen rows or the set of chosen columns is different.

Input

The input is given in the following format:

N K
$A_{1,1} A_{1,2} \ldots A_{1,N}$
$\vdots$
$A_{N,1} A_{N,2} \ldots A_{N,N}$

- $1 \le K < N \le 1000$
- $1 \le K \le 5$
- $1 \le A_{i,j} \le (N-K)^2$ $(1 \le i \le N, 1 \le j \le N)$
- All input values are integers.

Output

Print the number of ways to perform the operations, modulo 998244353.

Examples

standard input	standard output
3 1 1 2 4 3 4 2 2 1 3	6
6 5 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	36
7 5 2 3 1 4 1 3 1 4 4 1 4 1 1 1 1 3 4 1 4 3 4 4 2 2 2 1 2 2 2 4 1 4 2 3 1 2 3 1 3 1 2 4 3 3 2 1 4 2 2	36

Note

In the first example, there are 6 ways to perform the operations that satisfy the condition. For each of these, the remaining $(N-K)^2$ integers are exactly $1, 2, 3, 4$ in increasing order.

- Select row 1 in the first operation and column 1 in the second operation.
- Select row 1 in the first operation and column 3 in the second operation.
- Select row 2 in the first operation and column 1 in the second operation.
- Select row 2 in the first operation and column 2 in the second operation.
- Select row 3 in the first operation and column 2 in the second operation.
- Select row 3 in the first operation and column 3 in the second operation.

Problem M. Up-Down Sequence

Input file:	`standard input`
Output file:	`standard output`
Time limit:	2 seconds
Memory limit:	1024 megabytes

For a given integer N, determine whether there exists a permutation $P = (p_1, p_2, \ldots, p_N)$ of the integers $(1, 2, \ldots, N)$ that satisfies the following condition.

Let an index triple (i, j, k) be any triple of indices such that $1 \le i < j < k \le N$.

- The number of index triples (i, j, k) for which $p_i < p_j < p_k$ (a strictly increasing triple) is equal to the number of index triples (i, j, k) for which $p_i > p_j > p_k$ (a strictly decreasing triple).

If such a permutation exists, construct and output any one of them. If no such permutation exists, output -1.

You are given T test cases to solve.

Input

The input is given in the following format:

T
case_1
case_2
$\vdots$
case_T

Each test case is given in the following format:

N

- $1 \le T \le 5 \times 10^5$
- $3 \le N \le 5 \times 10^5$
- The sum of N over all test cases does not exceed 5×10^5.
- All input values are integers.

Output

For each test case, if no such permutation exists, print -1. Otherwise, print N space-separated integers representing the elements of the permutation P. If there are multiple such permutations, any of them will be accepted.

Example

standard input	standard output
2 3 4	1 3 2 2 4 1 3

Note

In the first test case, $N = 3$. One possible valid permutation is $(1, 3, 2)$. There is exactly one index triple $(1, 2, 3)$, whose corresponding values are $(1, 3, 2)$. This triple is neither strictly increasing nor strictly decreasing. Therefore, the number of increasing triples is 0 and the number of decreasing triples is 0, so the condition is satisfied.

In the second test case, $N = 4$. One possible valid permutation is $(2, 4, 1, 3)$. The index triples are

$$(1, 2, 3),\ (1, 2, 4),\ (1, 3, 4),\ \text{and}\ (2, 3, 4).$$

For each of these triples, the corresponding values are neither strictly increasing nor strictly decreasing. Thus, both the number of increasing triples and the number of decreasing triples are 0, and the condition is satisfied.