A. Max and Mod

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

You are given an integer n. Find any permutation p of length n^* such that:

- For all $2 \le i \le n$, $\max(p_{i-1}, p_i) \bmod i\,^\dagger = i - 1$ is satisfied.

If it is impossible to find such a permutation p, output -1.

---

*A permutation of length n is an array consisting of n distinct integers from 1 to n in arbitrary order. For example, $[2, 3, 1, 5, 4]$ is a permutation, but $[1, 2, 2]$ is not a permutation (2 appears twice in the array), and $[1, 3, 4]$ is also not a permutation ($n = 3$ but there is 4 in the array).

$^\dagger x \bmod y$ denotes the remainder from dividing x by y.

Input

Each test contains multiple test cases. The first line contains the number of test cases t ($1 \le t \le 99$). The description of the test cases follows.

The first line of each test case contains a single integer n ($2 \le n \le 100$).

Output

For each test case:

- If such a permutation p doesn't exist, output a single integer -1.
- Otherwise, output n integers $p_1, p_2, \ldots, p_n$ — the permutation p you've found. If there are multiple answers, output any of them.

Standard Input	Standard Output
4 2 3 4 5	-1 3 2 1 -1 1 5 2 3 4

Note

In the first test case, it is impossible to find such a permutation p, so you should output -1.

In the fourth test case, $p = [1, 5, 2, 3, 4]$ satisfies the condition:

- For $i = 2$, $\max(p_1, p_2) = 5$ and $5 \bmod 2 = 1$.
- For $i = 3$, $\max(p_2, p_3) = 5$ and $5 \bmod 3 = 2$.
- For $i = 4$, $\max(p_3, p_4) = 3$ and $3 \bmod 4 = 3$.
- For $i = 5$, $\max(p_4, p_5) = 4$ and $4 \bmod 5 = 4$.

B. MIN = GCD

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

You are given a positive integer sequence a of length n. Determine if it is possible to rearrange a such that there exists an integer i ($1 \le i < n$) satisfying

$$\min([a_1, a_2, \ldots, a_i]) = \gcd([a_{i+1}, a_{i+2}, \ldots, a_n]).$$

Here $\gcd(c)$ denotes the greatest common divisor of c, which is the maximum positive integer that divides all integers in c.

Input

Each test contains multiple test cases. The first line contains the number of test cases t ($1 \le t \le 10^4$). The description of the test cases follows.

The first line of each test case contains a single integer n ($2 \le n \le 10^5$).

The second line contains n integers $a_1, a_2, \ldots, a_n$ ($1 \le a_i \le 10^{18}$).

It is guaranteed that the sum of n over all test cases does not exceed 10^5.

Output

For each test case, output "Yes" if it is possible, and "No" otherwise.

You can output the answer in any case (upper or lower). For example, the strings "yEs", "yes", "Yes", and "YES" will be recognized as positive responses.

Standard Input	Standard Output
7 2 1 1 2 1 2 3 2 2 3 3 2 3 4 5 4 5 6 9 3 3 998244359987710471 99824435698771045 1000000007 6 1 1 4 5 1 4	Yes No Yes No Yes Yes Yes

Note

In the first test case, rearrange a to $[1, 1]$ and let $i = 1$, then $\min([1]) = \gcd([1])$.

In the second test case, it can be shown that it is impossible.

In the third test case, rearrange a to $[3, 2, 2]$ and let $i = 2$, then $\min([3, 2]) = \gcd([2])$.

In the fifth test case, rearrange a to $[3, 4, 5, 6, 9]$ and let $i = 3$, then $\min([3, 4, 5]) = \gcd([6, 9])$.

,

C. You Soared Afar With Grace

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

You are given a permutation a and b of length n^*. You can perform the following operation at most n times:

- Choose two indices i and j ($1 \le i, j \le n$, $i \neq j$), swap a_i with a_j, swap b_i with b_j.

Determine whether a and b can be reverses of each other after operations. In other words, for each $i = 1, 2, \ldots, n$, $a_i = b_{n+1-i}$.

If it is possible, output any valid sequence of operations. Otherwise, output -1.

---

*A permutation of length n is an array consisting of n distinct integers from 1 to n in arbitrary order. For example, $[2, 3, 1, 5, 4]$ is a permutation, but $[1, 2, 2]$ is not a permutation (2 appears twice in the array), and $[1, 3, 4]$ is also not a permutation ($n = 3$ but there is 4 in the array).

Input

Each test contains multiple test cases. The first line contains the number of test cases t ($1 \le t \le 10^4$). The description of the test cases follows.

The first line of each test case contains a single integer n ($2 \le n \le 2 \cdot 10^5$) — the length of the permutations.

The second line contains n integers $a_1, a_2, \ldots, a_n$ ($1 \le a_i \le n$).

The third line contains n integers $b_1, b_2, \ldots, b_n$ ($1 \le b_i \le n$).

It is guaranteed that a and b are permutations of length n.

It is guaranteed that the sum of n over all test cases does not exceed $2 \cdot 10^5$.

Output

For each test case, if it is impossible, output -1 in the only line. Otherwise, output a single integer m ($0 \le m \le n$) — the number of operations in the first line. In the following m lines, output two integers — the indices i and j ($1 \le i, j \le n$, $i \neq j$) in each operation in order. If there are multiple solutions, print any of them.

Standard Input	Standard Output
5 2 1 2 1 2 2 1 2 2 1 4 1 3 2 4 2 4 1 3 5 2 5 1 3 4 3 5 4 2 1 5 3 1 2 4 5 1 2 3 4 5	-1 0 1 1 2 2 1 2 1 3 -1

Note

In the second test case, b is already the reverse of a.

In the third test case, after performing the following operation, b will become the reverse of a:

- Swap a_1, a_2 and swap b_1, b_2. Now $a = [3, 1, 2, 4]$ and $b = [4, 2, 1, 3]$.

In the fourth test case, after performing the following operations in order, b will become the reverse of a:

- Swap a_1, a_2 and swap b_1, b_2. Now $a = [5, 2, 1, 3, 4]$ and $b = [5, 3, 4, 2, 1]$.
- Swap a_1, a_3 and swap b_1, b_3. Now $a = [1, 2, 5, 3, 4]$ and $b = [4, 3, 5, 2, 1]$.

D. Arcology On Permafrost

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

You are given three integers n, m, and k, where $m \cdot k < n$.

For a sequence b consisting of non-negative integers, define $f(b)$ as follows:

- You may perform the following operation on b:
  - Let l denote the current length of b. Choose a positive integer $1 \le i \le l - k + 1$, remove the subarray from index i to $i + k - 1$ and concatenate the remaining parts. In other words, replace b with

$$[b_1, b_2, \ldots, b_{i-1}, b_{i+k}, b_{i+k+1}, \ldots, b_l].$$

- $f(b)$ is defined as the **minimum** possible value of $\operatorname{mex}(b)^*$ after performing the above operation **at most** m **times** (possibly zero).

You need to construct a sequence a of length n consisting of non-negative integers, such that:

- For all $1 \le i \le n$, $0 \le a_i \le 10^9$.
- Over all such sequences a, $f(a)$ is **maximized**.

---

*The minimum excluded (MEX) of a collection of integers $c_1, c_2, \ldots, c_k$ is defined as the smallest non-negative integer x which does not occur in the collection c.

Input

Each test contains multiple test cases. The first line contains the number of test cases t ($1 \le t \le 10^4$). The description of the test cases follows.

The first line of each test case contains three integers n, m, and k ($2 \le n \le 2 \cdot 10^5$, $1 \le m < n$, $1 \le k < n$, $1 \le m \cdot k < n$).

It is guaranteed that the sum of n over all test cases does not exceed $2 \cdot 10^5$.

Output

For each test case, output n integers $a_1, a_2, \ldots, a_n$ ($0 \le a_i \le 10^9$).

If there are multiple answers, print any of them.

Standard Input	Standard Output
8	0 0
2 1 1	0 1 0 0 0
5 2 2	0 1 2 2 0 1
6 1 4	0 2 1 0 1 0 8 1
8 2 2	0 1 2 1000000000 1 0 1 2
8 1 5	1 0 0 1 0 2 1 0 2 1 0
11 3 3	0 2 1 0 2 1 0 3 2 1 0 2 1 0 2 1 0 2 1 0 2 1
22 6 3	4 0 2 1 3 4 0 2 1 0 3 4 0 1 2 1 3
17 2 2	

Note

In the first test case, it can be shown that $f(a) = 1$, which is maximized.

In the second test case, it can be shown that $f(a) = 1$, which is maximized. $f(a) = 1$ since you can perform the following operations:

- Choose $i = 3$, remove the subarray from index 3 to 4 and concatenate the remaining parts. The sequence a becomes $[0, 1, 0]$.
- Choose $i = 1$, remove the subarray from index 1 to 2 and concatenate the remaining parts. The sequence a becomes $[0]$.

In the third test case, it can be shown that $f(a) = 2$, which is maximized. $f(a) = 2$ since you can perform the following operation:

- Choose $i = 2$, remove the subarray from index 2 to 5 and concatenate the remaining parts. The sequence a becomes $[0, 1]$.

In the fourth test case, it can be shown that $f(a) = 2$, which is maximized.

In the fifth test case, it can be shown that $f(a) = 3$, which is maximized.

In the sixth test case, it can be shown that $f(a) = 2$, which is maximized.

,

E. Blossom

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

You are given a permutation a of length n^* where some elements are missing and represented by -1.

Define the value of a permutation as the sum of the MEX† of all its non-empty subsegments‡.

Find the sum of the value of all possible valid permutations that can be formed by filling in the missing elements of a modulo $10^9 + 7$.

---

* A permutation of length n is an array consisting of n distinct integers **from $\mathbf{0}$ to $\mathbf{n-1}$** in arbitrary order. For example, $[1, 2, 0, 4, 3]$ is a permutation, but $[0, 1, 1]$ is not a permutation (1 appears twice in the array), and $[0, 2, 3]$ is also not a permutation ($n = 3$ but there is 3 in the array).

† The minimum excluded (MEX) of a collection of integers $c_1, c_2, \ldots, c_k$ is defined as the smallest non-negative integer x which does not occur in the collection c.

‡ A sequence a is a subsegment of a sequence b if a can be obtained from b by the deletion of several (possibly, zero or all) elements from the beginning and several (possibly, zero or all) elements from the end.

Input

Each test contains multiple test cases. The first line contains the number of test cases t ($1 \le t \le 1000$). The description of the test cases follows.

The first line of each test case contains a single integer n ($1 \le n \le 5000$).

The second line contains n integers $a_1, a_2, \ldots, a_n$ ($-1 \le a_i < n$).

It is guaranteed that the elements of a that are not -1 are distinct.

It is guaranteed that the sum of n over all test cases does not exceed 5000.

Output

For each test case, output a single integer — the sum of the value of all possible valid permutations modulo $10^9 + 7$.

Standard Input	Standard Output
5 2 0 -1 2 -1 -1 3 2 0 1 3 -1 2 -1 5 -1 0 -1 2 -1	3 6 7 10 104

Note

In the first test case, the only valid permutation is $[0, 1]$, and the value of $[0, 1]$ is 3 since:

$$\operatorname{mex}([0]) + \operatorname{mex}([1]) + \operatorname{mex}([0, 1]) = 1 + 0 + 2 = 3$$

So the answer is 3.

In the second test case, there are two valid permutations: $[0, 1]$ and $[1, 0]$. The value of $[0, 1]$ and the value of $[1, 0]$ is 3, so the answer is $3 + 3 = 6$.

In the fourth test case, there are two valid permutations: $[0, 2, 1]$ and $[1, 2, 0]$. The value of $[0, 2, 1]$ is 5 since:

$$\operatorname{mex}([0]) + \operatorname{mex}([2]) + \operatorname{mex}([1]) + \operatorname{mex}([0, 2]) + \operatorname{mex}([2, 1]) + \operatorname{mex}([0, 2, 1]) = 1 + 0 + 0 + 1 + 0 + 3 = 5$$

And the value of $[1, 2, 0]$ is 5 since:

$$\operatorname{mex}([1]) + \operatorname{mex}([2]) + \operatorname{mex}([0]) + \operatorname{mex}([1, 2]) + \operatorname{mex}([2, 0]) + \operatorname{mex}([1, 2, 0]) = 0 + 0 + 1 + 0 + 1 + 3 = 5$$

So the answer is $5 + 5 = 10$.

F. Skyscape

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

You are given a permutation a of length n^*.

We say that a permutation b of length n is *good* if the two permutations a and b can become the same after performing the following operation **at most n times** (possibly zero):

- Choose two integers l, r such that $1 \le l < r \le n$ and $a_r = \min(a_l, a_{l+1}, \ldots, a_r)$.
- Cyclically shift the subsegment $[a_l, a_{l+1}, \ldots, a_r]$ one position to the right. In other words, replace a with $[a_1, \ldots, a_{l-1},\ a_r, a_l, a_{l+1}, \ldots, a_{r-1},\ a_{r+1}, \ldots, a_n]$.

You are also given a permutation c of length n where some elements are missing and represented by 0.

You need to find a **good** permutation $b_1, b_2, \ldots, b_n$ such that b can be formed by filling in the missing elements of c (i.e., for all $1 \le i \le n$, if $c_i \ne 0$, then $b_i = c_i$). If it is impossible, output -1.

---

*A permutation of length n is an array consisting of n distinct integers from 1 to n in arbitrary order. For example, $[2,3,1,5,4]$ is a permutation, but $[1,2,2]$ is not a permutation (2 appears twice in the array), and $[1,3,4]$ is also not a permutation ($n = 3$ but there is 4 in the array).

Input

Each test contains multiple test cases. The first line contains the number of test cases t ($1 \le t \le 10^4$). The description of the test cases follows.

The first line of each test case contains a single integer n ($2 \le n \le 5 \cdot 10^5$).

The second line of each test case contains n integers $a_1, a_2, \ldots, a_n$ ($1 \le a_i \le n$). It is guaranteed that a is a permutation of length n.

The third line of each test case contains n integers $c_1, c_2, \ldots, c_n$ ($0 \le c_i \le n$). It is guaranteed that the elements of c that are not 0 are distinct.

It is guaranteed that the sum of n over all test cases does not exceed $5 \cdot 10^5$.

Output

For each test case:

- If it is impossible to find such a good permutation b, output a single integer -1.
- Otherwise, output n integers $b_1, b_2, \ldots, b_n$ — the good permutation b you've found. You need to ensure that for all $1 \le i \le n$, if $c_i \ne 0$, then $b_i = c_i$. If there are multiple answers, output any of them.

Standard Input	Standard Output
9 2 2 1 1 2 4 3 2 4 1 2 0 0 1 5 3 2 1 5 4 1 3 0 0 0	1 2 2 3 4 1 1 3 2 4 5 3 2 1 5 4 -1 3 2 1 5 4 6 -1 -1 1 3 8 5 7 9 4 6 2

5 3 2 1 5 4 3 2 1 5 4 5 3 2 1 5 4 3 2 5 1 4 6 3 5 6 2 1 4 0 2 0 5 0 0 6 3 5 6 2 1 4 0 2 0 6 4 0 9 6 9 2 4 1 7 8 3 5 0 2 5 9 0 0 0 8 0 9 8 5 3 9 1 7 4 6 2 0 0 8 0 7 0 4 0 2	

Note

In the first test case, $b = [1, 2]$ is a valid answer since after performing the following operation, a and b will become the same:

- Choose $l = 1, r = 2$ and cyclically shift the subsegment $[a_1, a_2]$ one position to the right. Then a will become $[1, 2]$.

In the second test case, $b = [2, 3, 4, 1]$ is a valid answer since after performing the following operation, a and b will become the same:

- Choose $l = 1, r = 2$ and cyclically shift the subsegment $[a_1, a_2]$ one position to the right. Then a will become $[2, 3, 4, 1]$.

In the third test case, $b = [1, 3, 2, 4, 5]$ is a valid answer since after performing the following operation, a and b will become the same:

- Choose $l = 1, r = 3$ and cyclically shift the subsegment $[a_1, a_2, a_3]$ one position to the right. Then a will become $[1, 3, 2, 5, 4]$.
- Choose $l = 4, r = 5$ and cyclically shift the subsegment $[a_4, a_5]$ one position to the right. Then a will become $[1, 3, 2, 4, 5]$.

In the fourth test case, $b = [3, 2, 1, 5, 4]$ is a valid answer since a and b are already the same.

In the fifth test case, it is impossible to find such a good permutation b, so you should output -1.

In the sixth test case, $b = [3, 2, 1, 5, 4, 6]$ is a valid answer since after performing the following operation, a and b will become the same:

- Choose $l = 2, r = 4$ and cyclically shift the subsegment $[a_2, a_3, a_4]$ one position to the right. Then a will become $[3, 2, 5, 6, 1, 4]$.
- Choose $l = 3, r = 5$ and cyclically shift the subsegment $[a_3, a_4, a_5]$ one position to the right. Then a will become $[3, 2, 1, 5, 6, 4]$.
- Choose $l = 5, r = 6$ and cyclically shift the subsegment $[a_5, a_6]$ one position to the right. Then a will become $[3, 2, 1, 5, 4, 6]$.

G1. Wish Upon a Satellite (Easy Version)

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

This is the easy version of the problem. The difference between the versions is that in this version, $t \le 1000$, $n \le 5000$ and the sum of n does not exceed 5000. You can hack only if you solved all versions of this problem.

For a non-empty sequence c of length k, define $f(c)$ as follows:

- Turtle and Piggy are playing a game on a sequence. They are given the sequence $c_1, c_2, \ldots, c_k$, and Turtle goes first. Turtle and Piggy alternate in turns (so, Turtle does the first turn, Piggy does the second, Turtle does the third, etc.).
- The game goes as follows:
  - Let the current length of the sequence be m. If $m = 1$, the game ends.
  - If the game does not end and it's Turtle's turn, then Turtle must choose an integer i such that $1 \le i \le m - 1$, set c_i to $\min(c_i, c_{i+1})$, and remove c_{i+1}.
  - If the game does not end and it's Piggy's turn, then Piggy must choose an integer i such that $1 \le i \le m - 1$, set c_i to $\max(c_i, c_{i+1})$, and remove c_{i+1}.
- Turtle wants to **maximize** the value of c_1 in the end, while Piggy wants to **minimize** the value of c_1 in the end.
- $f(c)$ is the value of c_1 in the end if both players play optimally.

For a permutation p of length n^*, Turtle defines the *beauty* of the permutation as $\sum\limits_{i=1}^{n}\sum\limits_{j=i}^{n} f([p_i, p_{i+1}, \ldots, p_j])$ (i.e., the sum of $f(c)$ where c is a non-empty subsegment† of p).

Piggy gives Turtle a permutation a of length n where some elements are missing and represented by 0.

Turtle asks you to determine a permutation b of length n such that:

- b can be formed by filling in the missing elements of a (i.e., for all $1 \le i \le n$, if $a_i \ne 0$, then $b_i = a_i$).
- The beauty of the permutation b is **maximized**.

For convenience, you only need to find the maximum beauty of such permutation b.

---

*A permutation of length n is an array consisting of n distinct integers from 1 to n in arbitrary order. For example, $[2, 3, 1, 5, 4]$ is a permutation, but $[1, 2, 2]$ is not a permutation (2 appears twice in the array), and $[1, 3, 4]$ is also not a permutation ($n = 3$ but there is 4 in the array).

†A sequence a is a subsegment of a sequence b if a can be obtained from b by the deletion of several (possibly, zero or all) elements from the beginning and several (possibly, zero or all) elements from the end.

Input

Each test contains multiple test cases. The first line contains the number of test cases t ($1 \le t \le 1000$). The description of the test cases follows.

The first line of each test case contains a single integer n ($1 \le n \le 5000$).

The second line of each test case contains n integers $a_1, a_2, \ldots, a_n$ ($0 \le a_i \le n$). It is guaranteed that the elements of a that are not 0 are distinct.

It is guaranteed that the sum of n over all test cases does not exceed 5000.

Output

For each test case, output a single integer — the maximum beauty of the permutation b.

Standard Input	Standard Output
8 2 1 0 3 0 0 0 3 0 1 0 5 3 2 4 5 1 7 0 3 2 5 0 0 0 10 1 2 6 5 8 9 0 0 0 0 5 0 4 1 0 0 5 0 1 5 2 3	4 12 11 44 110 300 45 40

Note

In the first test case, the permutation b with the maximum beauty is $[1, 2]$. The beauty of $[1, 2]$ is 4 since $f([1]) + f([2]) + f([1, 2]) = 1 + 2 + 1 = 4$. If $c = [1, 2]$, then $f(c) = 1$ since Turtle can only choose $i = 1$ and he will set c_1 to $\min(c_1, c_2) = 1$.

In the second test case, one of the permutations b with the maximum beauty is $[3, 2, 1]$. The beauty of $[3, 2, 1]$ is 12 since $f([3]) + f([2]) + f([1]) + f([3, 2]) + f([2, 1]) + f([3, 2, 1]) = 3 + 2 + 1 + 2 + 1 + 3 = 12$.

In the third test case, one of the permutations b with the maximum beauty is $[2, 1, 3]$.

In the fourth test case, if $c = [3, 2, 4, 5, 1]$, then $f(c) = 3$. One of the possible game processes is as follows:

- Turtle can choose $i = 3$. Then he will set c_3 to $\min(c_3, c_4) = 4$ and remove c_4. The sequence c will become $[3, 2, 4, 1]$.
- Piggy can choose $i = 1$. Then he will set c_1 to $\max(c_1, c_2) = 3$ and remove c_2. The sequence c will become $[3, 4, 1]$.
- Turtle can choose $i = 2$. Then he will set c_2 to $\min(c_2, c_3) = 1$ and remove c_3. The sequence c will become $[3, 1]$.
- Piggy can choose $i = 1$. Then he will set c_1 to $\max(c_1, c_2) = 3$ and remove c_2. The sequence c will become $[3]$.
- The length of the sequence becomes 1, so the game will end. The value of c_1 will be 3 in the end.

In the fifth test case, one of the permutations b with the maximum beauty is $[1, 3, 2, 5, 6, 4, 7]$.

G2. Wish Upon a Satellite (Hard Version)

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

This is the hard version of the problem. The difference between the versions is that in this version, $t \le 10^4$, $n \le 5 \cdot 10^5$ and the sum of n does not exceed $5 \cdot 10^5$. You can hack only if you solved all versions of this problem.

For a non-empty sequence c of length k, define $f(c)$ as follows:

- Turtle and Piggy are playing a game on a sequence. They are given the sequence $c_1, c_2, \ldots, c_k$, and Turtle goes first. Turtle and Piggy alternate in turns (so, Turtle does the first turn, Piggy does the second, Turtle does the third, etc.).
- The game goes as follows:
  - Let the current length of the sequence be m. If $m = 1$, the game ends.
  - If the game does not end and it's Turtle's turn, then Turtle must choose an integer i such that $1 \le i \le m-1$, set c_i to $\min(c_i, c_{i+1})$, and remove c_{i+1}.
  - If the game does not end and it's Piggy's turn, then Piggy must choose an integer i such that $1 \le i \le m-1$, set c_i to $\max(c_i, c_{i+1})$, and remove c_{i+1}.
- Turtle wants to **maximize** the value of c_1 in the end, while Piggy wants to **minimize** the value of c_1 in the end.
- $f(c)$ is the value of c_1 in the end if both players play optimally.

For a permutation p of length n^*, Turtle defines the *beauty* of the permutation as $\sum_{i=1}^{n}\sum_{j=i}^{n} f([p_i, p_{i+1}, \ldots, p_j])$ (i.e., the sum of $f(c)$ where c is a non-empty subsegment† of p).

Piggy gives Turtle a permutation a of length n where some elements are missing and represented by 0.

Turtle asks you to determine a permutation b of length n such that:

- b can be formed by filling in the missing elements of a (i.e., for all $1 \le i \le n$, if $a_i \ne 0$, then $b_i = a_i$).
- The beauty of the permutation b is **maximized**.

For convenience, you only need to find the maximum beauty of such permutation b.

---

*A permutation of length n is an array consisting of n distinct integers from 1 to n in arbitrary order. For example, $[2,3,1,5,4]$ is a permutation, but $[1,2,2]$ is not a permutation (2 appears twice in the array), and $[1,3,4]$ is also not a permutation ($n=3$ but there is 4 in the array).

†A sequence a is a subsegment of a sequence b if a can be obtained from b by the deletion of several (possibly, zero or all) elements from the beginning and several (possibly, zero or all) elements from the end.

Input

Each test contains multiple test cases. The first line contains the number of test cases t ($1 \le t \le 10^4$). The description of the test cases follows.

The first line of each test case contains a single integer n ($1 \le n \le 5 \cdot 10^5$).

The second line of each test case contains n integers $a_1, a_2, \ldots, a_n$ ($0 \le a_i \le n$). It is guaranteed that the elements of a that are not 0 are distinct.

It is guaranteed that the sum of n over all test cases does not exceed $5 \cdot 10^5$.

Output

For each test case, output a single integer — the maximum beauty of the permutation b.

Standard Input	Standard Output
8 2 1 0 3 0 0 0 3 0 1 0 5 3 2 4 5 1 7 0 3 2 5 0 0 0 10 1 2 6 5 8 9 0 0 0 0 5 0 4 1 0 0 5 0 1 5 2 3	4 12 11 44 110 300 45 40

Note

In the first test case, the permutation b with the maximum beauty is $[1, 2]$. The beauty of $[1, 2]$ is 4 since $f([1]) + f([2]) + f([1, 2]) = 1 + 2 + 1 = 4$. If $c = [1, 2]$, then $f(c) = 1$ since Turtle can only choose $i = 1$ and he will set c_1 to $\min(c_1, c_2) = 1$.

In the second test case, one of the permutations b with the maximum beauty is $[3, 2, 1]$. The beauty of $[3, 2, 1]$ is 12 since $f([3]) + f([2]) + f([1]) + f([3, 2]) + f([2, 1]) + f([3, 2, 1]) = 3 + 2 + 1 + 2 + 1 + 3 = 12$.

In the third test case, one of the permutations b with the maximum beauty is $[2, 1, 3]$.

In the fourth test case, if $c = [3, 2, 4, 5, 1]$, then $f(c) = 3$. One of the possible game processes is as follows:

- Turtle can choose $i = 3$. Then he will set c_3 to $\min(c_3, c_4) = 4$ and remove c_4. The sequence c will become $[3, 2, 4, 1]$.
- Piggy can choose $i = 1$. Then he will set c_1 to $\max(c_1, c_2) = 3$ and remove c_2. The sequence c will become $[3, 4, 1]$.
- Turtle can choose $i = 2$. Then he will set c_2 to $\min(c_2, c_3) = 1$ and remove c_3. The sequence c will become $[3, 1]$.
- Piggy can choose $i = 1$. Then he will set c_1 to $\max(c_1, c_2) = 3$ and remove c_2. The sequence c will become $[3]$.
- The length of the sequence becomes 1, so the game will end. The value of c_1 will be 3 in the end.

In the fifth test case, one of the permutations b with the maximum beauty is $[1, 3, 2, 5, 6, 4, 7]$.

,

H. Turtle and Nediam 2

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

LGR-205-Div.1 C Turtle and Nediam

You are given a binary sequence s of length n which only consists of 0 and 1.

You can do the following operation **at most $n-2$ times** (possibly zero):

- Let m denote the current length of s. Choose an integer i such that $1 \le i \le m-2$.
- Let the median* of the subarray $[s_i, s_{i+1}, s_{i+2}]$ be x, and let j be the smallest integer such that $j \ge i$ and $s_j = x$.
- Remove s_j from the sequence and concatenate the remaining parts. In other words, replace s with $[s_1, s_2, \ldots, s_{j-1}, s_{j+1}, s_{j+2}, \ldots, s_m]$.

Note that after every operation, the length of s decreases by 1.

Find how many different binary sequences can be obtained after performing the operation, modulo 10^9+7.

*The median of an array of odd length k is the $\frac{k+1}{2}$-th element when sorted.

Input

Each test contains multiple test cases. The first line contains the number of test cases t ($1 \le t \le 10^4$). The description of the test cases follows.

The first line of each test case contains a single integer n ($3 \le n \le 2 \cdot 10^6$) — the length of the binary sequence.

The second line of each test case contains a string s of length n, consisting of only 0 and 1.

It is guaranteed that the sum of n over all test cases does not exceed $2 \cdot 10^6$.

Output

For each test case, output a single integer — the number of binary sequences that can be obtained, modulo 10^9+7.

Standard Input	Standard Output
5 5 11111 6 100011 9 000111000 14 11001111111000 16 0010000110100011	4 8 30 114 514

Note

In the first test case, the following binary sequences can be obtained: $[1,1]$, $[1,1,1]$, $[1,1,1,1]$, $[1,1,1,1,1]$.

In the second test case, the following binary sequences can be obtained: $[0,1]$, $[0,1,1]$, $[1,0,1]$, $[1,0,0,1]$, $[1,0,1,1]$, $[1,0,0,0,1]$, $[1,0,0,1,1]$, $[1,0,0,0,1,1]$. For example, to obtain $[0,1,1]$, you can:

- Choose $i = 2$. The median of $[0, 0, 0]$ is 0. Remove s_2. The sequence becomes $[1, 0, 0, 1, 1]$.
- Choose $i = 1$. The median of $[1, 0, 0]$ is 0. Remove s_2. The sequence becomes $[1, 0, 1, 1]$.
- Choose $i = 1$. The median of $[1, 0, 1]$ is 1. Remove s_1. The sequence becomes $[0, 1, 1]$.