A. Nene's Game

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

Nene invented a new game based on an increasing sequence of integers $a_1, a_2, \ldots, a_k$.

In this game, initially n players are lined up in a row. In each of the rounds of this game, the following happens:

- Nene finds the a_1-th, a_2-th, $\ldots$, a_k-th players in a row. They are kicked out of the game simultaneously. If the i-th player in a row should be kicked out, but there are fewer than i players in a row, they are skipped.

Once no one is kicked out of the game in some round, all the players that are still in the game are declared as winners.

For example, consider the game with $a = [3, 5]$ and $n = 5$ players. Let the players be named player A, player B, $\ldots$, player E in the order they are lined up initially. Then,

- Before the first round, players are lined up as ABCDE. Nene finds the 3-rd and the 5-th players in a row. These are players C and E. They are kicked out in the first round.
- Now players are lined up as ABD. Nene finds the 3-rd and the 5-th players in a row. The 3-rd player is player D and there is no 5-th player in a row. Thus, only player D is kicked out in the second round.
- In the third round, no one is kicked out of the game, so the game ends after this round.
- Players A and B are declared as the winners.

Nene has not yet decided how many people would join the game initially. Nene gave you q integers $n_1, n_2, \ldots, n_q$ and you should answer the following question for each $1 \le i \le q$ **independently**:

- How many people would be declared as winners if there are n_i players in the game initially?

Input

Each test contains multiple test cases. The first line contains the number of test cases t ($1 \le t \le 250$). The description of test cases follows.

The first line case contains two integers k and q ($1 \le k, q \le 100$) — the length of the sequence a and the number of values n_i you should solve this problem for.

The second line contains k integers $a_1, a_2, \ldots, a_k$ ($1 \le a_1 < a_2 < \ldots < a_k \le 100$) — the sequence a.

The third line contains q integers $n_1, n_2, \ldots, n_q$ ($1 \le n_i \le 100$).

Output

For each test case, output q integers: the i-th ($1 \le i \le q$) of them should be the number of players declared as winners if initially n_i players join the game.

Standard Input	Standard Output
6 2 1 3 5 5	2 1 1 1 1 2 2 2 1 10 68

5 3 2 4 6 7 9 1 3 5 5 4 3 4 5 6 7 1 2 3 4 2 3 69 96 1 10 100 1 1 100 50 3 3 10 20 30 1 10 100	50 1 9 9

Note

The first test case was explained in the statement.

In the second test case, when $n = 1$, the only player stays in the game in the first round. After that, the game ends and the only player is declared as a winner.

,

B. Nene and the Card Game

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

You and Nene are playing a card game. The deck with $2n$ cards is used to play this game. Each card has an integer from 1 to n on it, and each of integers 1 through n appears exactly on 2 cards. Additionally, there is a table where cards are placed during the game (initially, the table is empty).

In the beginning of the game, these $2n$ cards are distributed between you and Nene so that each player receives n cards.

After it, you and Nene alternatively take $2n$ turns, i.e. each person takes n turns, **starting with you**. On each turn:

- The player whose turn is it selects one of the cards in his hand. Let x be the number on it.
- The player whose turn is it receives 1 point if there is already a card with the integer x on the table (otherwise, he receives no points). After it, he places the selected card with the integer x on the table.

Note that turns are made publicly: each player can see all the cards on the table at each moment.

Nene is very smart so she always selects cards optimally in order to maximize her score in the end of the game (after $2n$ rounds). If she has several optimal moves, she selects the move that minimizes your score in the end of the game.

More formally, Nene always takes turns optimally in order to maximize her score in the end of the game in the first place and to minimize your score in the end of the game in the second place.

Assuming that the cards are already distributed and cards in your hand have integers $a_1, a_2, \ldots, a_n$ written on them, what is the maximum number of points you can get by taking your turns optimally?

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 test cases follows.

The first line contains a single integer n ($1 \le n \le 2 \cdot 10^5$) — the number of cards you and Nene receive in the beginning of the game.

The second line contains n integers $a_1, a_2, \ldots, a_n$ ($1 \le a_i \le n$) — the integers on the cards in your hand. It is guaranteed that each integer from 1 through n appears in the sequence $a_1, a_2, \ldots, a_n$ at most 2 times.

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 one integer: the maximum number of points you can get.

Standard Input	Standard Output
5 4 1 1 2 3 8	1 2 1

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

Note

In the first test case, the integers written on your cards are 1, 1, 2 and 3. The integers written on Nene's cards are 2, 3, 4 and 4. The game may proceed as follows:

1. You select one of the cards with an integer 1 written on it and place it on the table.
2. Nene selects one of the cards with an integer 4 written on it and places it on the table.
3. You select the card with an integer 1 written on it, receive 1 point, and place the selected card on the table.
4. Nene selects the card with an integer 4 written on it, receive 1 point, and places the selected card on the table.
5. You select the card with an integer 2 written on it and place it on the table.
6. Nene selects the card with an integer 2 written on it, receive 1 point, and places the selected card on the table.
7. You select the card with an integer 3 written on it and place it on the table.
8. Nene selects the card with an integer 3 written on it, receive 1 point, and places the selected card on the table.

At the end of the game, you scored 1 point, and Nene scored 3. It can be shown that you cannot score more than 1 point if Nene plays optimally, so the answer is 1.

In the second test case, if both players play optimally, you score 2 points and Nene scores 6 points.

C. Nene's Magical Matrix

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

The magical girl Nene has an $n \times n$ matrix a filled with zeroes. The j-th element of the i-th row of matrix a is denoted as $a_{i,j}$.

She can perform operations of the following two types with this matrix:

- Type 1 operation: choose an integer i between 1 and n and a permutation $p_1, p_2, \ldots, p_n$ of integers from 1 to n. Assign $a_{i,j} := p_j$ for all $1 \le j \le n$ simultaneously.
- Type 2 operation: choose an integer i between 1 and n and a permutation $p_1, p_2, \ldots, p_n$ of integers from 1 to n. Assign $a_{j,i} := p_j$ for all $1 \le j \le n$ simultaneously.

Nene wants to maximize the sum of all the numbers in the matrix $\sum\limits_{i=1}^{n}\sum\limits_{j=1}^{n} a_{i,j}$. She asks you to find the way to perform the operations so that this sum is maximized. As she doesn't want to make too many operations, you should provide a solution with no more than $2n$ operations.

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 ($1 \le t \le 500$). The description of test cases follows.

The only line of each test case contains a single integer n ($1 \le n \le 500$) — the size of the matrix a.

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

Output

For each test case, in the first line output two integers s and m ($0 \le m \le 2n$) — the maximum sum of the numbers in the matrix and the number of operations in your solution.

In the k-th of the next m lines output the description of the k-th operation:

- an integer c ($c \in \{1, 2\}$) — the type of the k-th operation;
- an integer i ($1 \le i \le n$) — the row or the column the k-th operation is applied to;
- a permutation $p_1, p_2, \ldots, p_n$ of integers from 1 to n — the permutation used in the k-th operation.

Note that you don't need to minimize the number of operations used, you only should use no more than $2n$ operations. It can be shown that the maximum possible sum can always be obtained in no more than $2n$ operations.

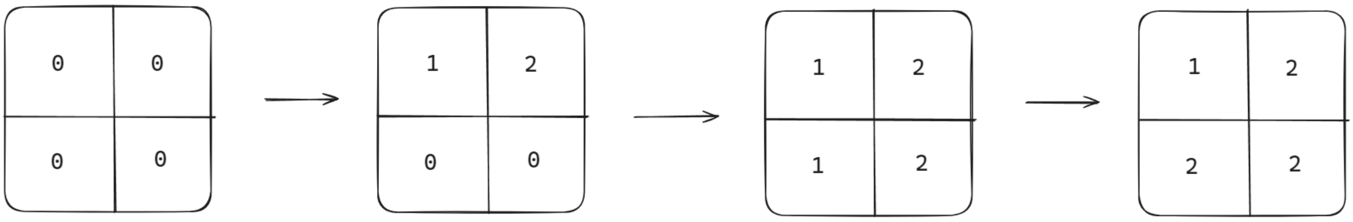
Standard Input	Standard Output
2 1 2	1 1 1 1 1 7 3

	1 1 1 2 1 2 1 2 2 1 1 2

Note

In the first test case, the maximum sum $s = 1$ can be obtained in 1 operation by setting $a_{1,1} := 1$.

In the second test case, the maximum sum $s = 7$ can be obtained in 3 operations as follows:



It can be shown that it is impossible to make the sum of the numbers in the matrix larger than 7.

,

D. Nene and the Mex Operator

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

Nene gave you an array of integers $a_1, a_2, \ldots, a_n$ of length n.

You can perform the following operation no more than $5 \cdot 10^5$ times (possibly zero):

- Choose two integers l and r such that $1 \le l \le r \le n$, compute x as $\operatorname{MEX}(\{a_l, a_{l+1}, \ldots, a_r\})$, and simultaneously set $a_l := x, a_{l+1} := x, \ldots, a_r := x$.

Here, MEX of a set of integers $\{c_1, c_2, \ldots, c_k\}$ is defined as the smallest non-negative integer m which does not occur in the set c.

Your goal is to maximize the sum of the elements of the array a. Find the maximum sum and construct a sequence of operations that achieves this sum. Note that you don't need to minimize the number of operations in this sequence, you only should use no more than $5 \cdot 10^5$ operations in your solution.

Input

The first line contains an integer n ($1 \le n \le 18$) — the length of the array a.

The second line contains n integers $a_1, a_2, \ldots, a_n$ ($0 \le a_i \le 10^7$) — the array a.

Output

In the first line, output two integers s and m ($0 \le m \le 5 \cdot 10^5$) — the maximum sum of elements of the array a and the number of operations in your solution.

In the i-th of the following m lines, output two integers l and r ($1 \le l \le r \le n$), representing the parameters of the i-th operation.

It can be shown that the maximum sum of elements of the array a can always be obtained in no more than $5 \cdot 10^5$ operations.

Standard Input	Standard Output
2 0 1	4 1 1 2
3 1 3 9	13 0
4 1 100 2 1	105 2 3 3 3 4
1 0	1 1 1 1

Note

In the first example, after the operation with $l = 1$ and $r = 2$ the array a becomes equal to $[2, 2]$. It can be shown that it is impossible to achieve a larger sum of the elements of a, so the answer is 4.

In the second example, the initial sum of elements is 13 which can be shown to be the largest.

In the third example, the array a changes as follows:

- after the first operation ($l = 3$, $r = 3$), the array a becomes equal to $[1, 100, 0, 1]$;
- after the second operation ($l = 3$, $r = 4$), the array a becomes equal to $[1, 100, 2, 2]$.

It can be shown that it is impossible to achieve a larger sum of the elements of a, so the answer is 105.

,

E1. Nene vs. Monsters (Easy Version)

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

This is the easy version of the problem. The only difference between the versions is the constraints on a_i. You can make hacks only if both versions of the problem are solved.

Nene is fighting with n monsters, located in a circle. These monsters are numbered from 1 to n, and the i-th ($1 \le i \le n$) monster's current energy level is a_i.

Since the monsters are too strong, Nene decided to fight with them using the `Attack Your Neighbour` spell. When Nene uses this spell, the following actions happen in the following order **one by one**:

- The 1-st monster attacks the 2-nd monster;
- The 2-nd monster attacks the 3-rd monster;
- $\ldots$
- The $(n-1)$-th monster attacks the n-th monster;
- The n-th monster attacks the 1-st monster.

When the monster with energy level x attacks the monster with the energy level y, the energy level of the defending monster becomes $\max(0, y - x)$ (the energy level of the attacking monster remains equal to x).

Nene is going to use this spell 10^{100} times and deal with the monsters that will still have a non-zero energy level herself. She wants you to determine which monsters will have a non-zero energy level once she will use the described spell 10^{100} times.

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 test cases follows.

The first line contains a single integer n ($2 \le n \le 2 \cdot 10^5$) — the number of monsters.

The second line contains n integers $a_1, a_2, \ldots, a_n$ ($0 \le a_i \le 2 \cdot 10^5$) — the current energy levels of monsters.

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

Output

For each test case,

- in the first line output an integer m — the number of monsters with non-zero energy level after 10^{100} uses of the spell;
- in the second line of output m integers $i_1, i_2, \ldots, i_m$ ($1 \le i_1 < i_2 < \ldots < i_m \le n$) — the indices of these monsters in the increasing order.

If $m = 0$, you may either output an empty line or don't output it.

Standard Input	Standard Output

5 3 2 5 3 2 0 0 4 1 5 7 2 4 4 2 1 2 13 1 1 4 5 1 4 1 9 1 9 8 1 0	1 1 0 1 1 2 1 3 6 1 3 6 8 10 12

Note

In the first test case, the following actions happen during the first 3 uses of the spell in this order:

- Nene uses the `Attack Your Neighbour` spell for the first time;
- the 1-st monster attacks the 2-nd monster, after the attack the energy level of the 2-nd monster becomes equal to $\max(0, 5 - 2) = 3$;
- the 2-nd monster attacks the 3-rd monster, after the attack the energy level of the 3-rd monster becomes equal to $\max(0, 3 - 3) = 0$;
- the 3-rd monster attacks the 1-st monster, after the attack the energy level of the 1-st monster becomes equal to $\max(0, 2 - 0) = 2$;
- Nene uses the `Attack Your Neighbour` spell for the second time;
- the 1-st monster attacks the 2-nd monster, after the attack the energy level of the 2-nd monster becomes equal to $\max(0, 3 - 2) = 1$;
- the 2-nd monster attacks the 3-rd monster, after the attack the energy level of the 3-rd monster becomes equal to $\max(0, 0 - 1) = 0$;
- the 3-rd monster attacks the 1-st monster, after the attack the energy level of the 1-st monster becomes equal to $\max(0, 2 - 0) = 2$;
- Nene uses the `Attack Your Neighbour` spell for the third time;
- the 1-st monster attacks the 2-nd monster, after the attack the energy level of the 2-nd monster becomes equal to $\max(0, 1 - 2) = 0$;
- the 2-nd monster attacks the 3-rd monster, after the attack the energy level of the 3-rd monster becomes equal to $\max(0, 0 - 0) = 0$;
- the 3-rd monster attacks the 1-st monster, after the attack the energy level of the 1-st monster becomes equal to $\max(0, 2 - 0) = 2$.

After each of the next uses of the spell, energy levels of monsters do not change. Thus, only the 1-st monster has a non-zero energy level in the end.

In the second test case, both monsters initially have zero energy level.

,

E2. Nene vs. Monsters (Hard Version)

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

This is the hard version of the problem. The only difference between the versions is the constraints on a_i. You can make hacks only if both versions of the problem are solved.

Nene is fighting with n monsters, located in a circle. These monsters are numbered from 1 to n, and the i-th ($1 \le i \le n$) monster's current energy level is a_i.

Since the monsters are too strong, Nene decided to fight with them using the `Attack Your Neighbour` spell. When Nene uses this spell, the following actions happen in the following order **one by one**:

- The 1-st monster attacks the 2-nd monster;
- The 2-nd monster attacks the 3-rd monster;
- $\ldots$
- The $(n-1)$-th monster attacks the n-th monster;
- The n-th monster attacks the 1-st monster.

When the monster with energy level x attacks the monster with the energy level y, the energy level of the defending monster becomes $\max(0, y-x)$ (the energy level of the attacking monster remains equal to x).

Nene is going to use this spell 10^{100} times and deal with the monsters that will still have a non-zero energy level herself. She wants you to determine which monsters will have a non-zero energy level once she will use the described spell 10^{100} times.

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 test cases follows.

The first line contains a single integer n ($2 \le n \le 2 \cdot 10^5$) — the number of monsters.

The second line contains n integers $a_1, a_2, \ldots, a_n$ $(0 \le a_i \le 10^9)$ — the current energy levels of monsters.

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

Output

For each test case,

- in the first line output an integer m — the number of monsters with non-zero energy level after 10^{100} uses of the spell;
- in the second line of output m integers $i_1, i_2, \ldots, i_m$ $(1 \le i_1 < i_2 < \ldots < i_m \le n)$ — the indices of these monsters in the increasing order.

If $m = 0$, you may either output an empty line or don't output it.

Standard Input	Standard Output
5 3	1 1

2 5 3	0
2	
0 0	1
4	1
1 5 7 2	2
4	1 3
4 2 1 2	6
13	1 3 6 8 10 12
1 1 4 5 1 4 1 9 1 9 8 1 0	

Note

In the first test case, the following actions happen during the first 3 uses of the spell in this order:

- Nene uses the `Attack Your Neighbour` spell for the first time;
- the 1-st monster attacks the 2-nd monster, after the attack the energy level of the 2-nd monster becomes equal to $\max(0, 5-2) = 3$;
- the 2-nd monster attacks the 3-rd monster, after the attack the energy level of the 3-rd monster becomes equal to $\max(0, 3-3) = 0$;
- the 3-rd monster attacks the 1-st monster, after the attack the energy level of the 1-st monster becomes equal to $\max(0, 2-0) = 2$;
- Nene uses the `Attack Your Neighbour` spell for the second time;
- the 1-st monster attacks the 2-nd monster, after the attack the energy level of the 2-nd monster becomes equal to $\max(0, 3-2) = 1$;
- the 2-nd monster attacks the 3-rd monster, after the attack the energy level of the 3-rd monster becomes equal to $\max(0, 0-1) = 0$;
- the 3-rd monster attacks the 1-st monster, after the attack the energy level of the 1-st monster becomes equal to $\max(0, 2-0) = 2$;
- Nene uses the `Attack Your Neighbour` spell for the third time;
- the 1-st monster attacks the 2-nd monster, after the attack the energy level of the 2-nd monster becomes equal to $\max(0, 1-2) = 0$;
- the 2-nd monster attacks the 3-rd monster, after the attack the energy level of the 3-rd monster becomes equal to $\max(0, 0-0) = 0$;
- the 3-rd monster attacks the 1-st monster, after the attack the energy level of the 1-st monster becomes equal to $\max(0, 2-0) = 2$.

After each of the next uses of the spell, energy levels of monsters do not change. Thus, only the 1-st monster has a non-zero energy level in the end.

In the second test case, both monsters initially have zero energy level.

F. Nene and the Passing Game

Input file:	standard input
Output file:	standard output
Time limit:	4 seconds
Memory limit:	256 megabytes

Nene is training her team as a basketball coach. Nene's team consists of n players, numbered from 1 to n. The i-th player has an *arm interval* $[l_i, r_i]$. Two players i and j ($i \neq j$) can pass the ball to each other if and only if $|i - j| \in [l_i + l_j, r_i + r_j]$ (here, $|x|$ denotes the absolute value of x).

Nene wants to test the cooperation ability of these players. In order to do this, she will hold several rounds of assessment.

- In each round, Nene will select a sequence of players $p_1, p_2, \ldots, p_m$ such that players p_i and p_{i+1} can pass the ball to each other for all $1 \le i < m$. The length of the sequence m can be chosen by Nene. Each player can appear in the sequence $p_1, p_2, \ldots, p_m$ multiple times or not appear in it at all.
- Then, Nene will throw a ball to player p_1, player p_1 will pass the ball to player p_2 and so on... Player p_m will throw a ball away from the basketball court so it can no longer be used.

As a coach, Nene wants each of n players to appear in at least one round of assessment. Since Nene has to go on a date after school, Nene wants you to calculate the minimum number of rounds of assessment needed to complete the task.

Input

Each test contains multiple test cases. The first line contains the number of test cases t ($1 \le t \le 2 \cdot 10^5$). The description of test cases follows.

The first line contains a single integer n ($1 \le n \le 2 \cdot 10^6$) — the number of players.

The i-th of the next n lines contains two integers l_i and r_i ($1 \le l_i \le r_i \le n$) — the arm interval of the i-th player.

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 one integer — the minimum number of rounds of assessment Nene needs to complete her work.

Standard Input	Standard Output
5 2 1 1 1 1 2 1 1 2 2 3 1 3 1 3 1 3	2 2 2 1 3

5 1 1 2 2 1 5 2 2 1 1 6 1 2 5 5 2 3 2 3 2 2 1 2	

Note

In the first two test cases, Nene can host two rounds of assessment: one with $p = [1]$ and one with $p = [2]$. It can be shown that hosting one round of assessment is not enough, so the answer is 2.

In the third test case, Nene can host two rounds of assessment: one with $p = [1, 3]$ and one with $p = [2]$. Player 1 can pass the ball to player 3 as $|3 - 1| = 2 \in [1 + 1, 3 + 3]$. It can be shown that hosting one round of assessment is not enough, so the answer is 2.