Codeforces Beta Round 6 (Div. 2 Only)

A. Triangle

2 seconds, 64 megabytes

Johnny has a younger sister Anne, who is very clever and smart. As she came home from the kindergarten, she told his brother about the task that her kindergartener asked her to solve. The task was just to construct a triangle out of four sticks of different colours. Naturally, one of the sticks is extra. It is not allowed to break the sticks or use their partial length. Anne has perfectly solved this task, now she is asking Johnny to do the same.

The boy answered that he would cope with it without any difficulty. However, after a while he found out that different tricky things can occur. It can happen that it is impossible to construct a triangle of a positive area, but it is possible to construct a degenerate triangle. It can be so, that it is impossible to construct a degenerate triangle even. As Johnny is very lazy, he does not want to consider such a big amount of cases, he asks you to help him.

Input
The first line of the input contains four space-separated positive integer numbers not exceeding 100 — lengthes of the sticks.

Output
Output `TRIANGLE` if it is possible to construct a non-degenerate triangle. Output `SEGMENT` if the first case cannot take place and it is possible to construct a degenerate triangle. Output `IMPOSSIBLE` if it is impossible to construct any triangle. Remember that you are to use three sticks. It is not allowed to break the sticks or use their partial length.

input
4 2 1 3
output
TRIANGLE

input
7 2 2 4
output
SEGMENT

input
3 5 9 1
output
IMPOSSIBLE

B. President's Office

2 seconds, 64 megabytes

President of Berland has a very vast office-room, where, apart from him, work his subordinates. Each subordinate, as well as President himself, has his own desk of a unique colour. Each desk is rectangular, and its sides are parallel to the office walls. One day President decided to establish an assembly, of which all his deputies will be members. Unfortunately, he does not remember the exact amount of his deputies, but he remembers that the desk of each his deputy is adjacent to his own desk, that is to say, the two desks (President's and each deputy's) have a common side of a positive length.

The office-room plan can be viewed as a matrix with n rows and m columns. Each cell of this matrix is either empty, or contains a part of a desk. An uppercase Latin letter stands for each desk colour. The «period» character («.») stands for an empty cell.

Input
The first line contains two separated by a space integer numbers n, m $(1 \le n, m \le 100)$ — the length and the width of the office-room, and c character — the President's desk colour. The following n lines contain m characters each — the office-room description. It is guaranteed that the colour of each desk is unique, and each desk represents a continuous subrectangle of the given matrix. All colours are marked by uppercase Latin letters.

Output
Print the only number — the amount of President's deputies.

input
3 4 R G.B. .RR. TTT.
output
2

input
3 3 Z ... .H. ..Z
output
0

C. Alice, Bob and Chocolate

2 seconds, 64 megabytes

Alice and Bob like games. And now they are ready to start a new game. They have placed n chocolate bars in a line. Alice starts to eat chocolate bars one by one from left to right, and Bob — from right to left. For each chocololate bar the time, needed for the player to consume it, is known (Alice and Bob eat them with equal speed). When the player consumes a chocolate bar, he immediately starts with another. It is not allowed to eat two chocolate bars at the same time, to leave the bar unfinished and to make pauses. If both players start to eat the same bar simultaneously, Bob leaves it to Alice as a true gentleman.

How many bars each of the players will consume?

Input
The first line contains one integer n $(1 \le n \le 10^5)$ — the amount of bars on the table. The second line contains a sequence $t_1, t_2, ..., t_n$ $(1 \le t_i \le 1000)$, where t_i is the time (in seconds) needed to consume the i-th bar (in the order from left to right).

Output
Print two numbers a and b, where a is the amount of bars consumed by Alice, and b is the amount of bars consumed by Bob.

input
5 2 9 8 2 7
output
2 3

D. Lizards and Basements 2

2 seconds, 64 megabytes

This is simplified version of the problem used on the original contest. The original problem seems to have too difiicult solution. The constraints for input data have been reduced.

Polycarp likes to play computer role-playing game «Lizards and Basements». At the moment he is playing it as a magician. At one of the last levels he has to fight the line of archers. The only spell with which he can damage them is a fire ball. If Polycarp hits the i-th archer with his fire ball (they are numbered from left to right), the archer loses a health points. At the same time the spell damages the archers adjacent to the i-th (if any) — they lose b $(1 \le b < a \le 10)$ health points each.

As the extreme archers (i.e. archers numbered 1 and n) are very far, the fire ball cannot reach them. Polycarp can hit any other archer with his fire ball.

The amount of health points for each archer is known. An archer will be killed when this amount is less than 0. What is the minimum amount of spells Polycarp can use to kill all the enemies?

Polycarp can throw his fire ball into an archer if the latter is already killed.

Input

The first line of the input contains three integers n, a, b $(3 \le n \le 10$; $1 \le b < a \le 10)$. The second line contains a sequence of n integers — $h_1, h_2, ..., h_n$ $(1 \le h_i \le 15)$, where h_i is the amount of health points the i-th archer has.

Output

In the first line print t — the required minimum amount of fire balls.

In the second line print t numbers — indexes of the archers that Polycarp should hit to kill all the archers in t shots. All these numbers should be between 2 and n - 1. Separate numbers with spaces. If there are several solutions, output any of them. Print numbers in any order.

input
3 2 1 2 2 2
output
3 2 2 2

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

E. Exposition

1.5 seconds, 64 megabytes

There are several days left before the fiftieth birthday of a famous Berland's writer Berlbury. In this connection the local library decided to make an exposition of the works of this famous science-fiction writer. It was decided as well that it is necessary to include into the exposition only those books that were published during a particular time period. It is obvious that if the books differ much in size, the visitors will not like it. That was why the organizers came to the opinion, that the difference between the highest and the lowest books in the exposition should be not more than k millimeters.

The library has n volumes of books by Berlbury, arranged in chronological order of their appearance. The height of each book in millimeters is know, it is h_i. As Berlbury is highly respected in the city, the organizers want to include into the exposition as many books as possible, and to find out what periods of his creative work they will manage to cover. You are asked to help the organizers cope with this hard task.

Input

The first line of the input data contains two integer numbers separated by a space n $(1 \le n \le 10^5)$ and k $(0 \le k \le 10^6)$ — the amount of books by Berlbury in the library, and the maximum allowed height difference between the lowest and the highest books. The second line contains n integer numbers separated by a space. Each number h_i $(1 \le h_i \le 10^6)$ is the height of the i-th book in millimeters.

Output

In the first line of the output data print two numbers a and b (separate them by a space), where a is the maximum amount of books the organizers can include into the exposition, and b — the amount of the time periods, during which Berlbury published a books, and the height difference between the lowest and the highest among these books is not more than k milllimeters.

In each of the following b lines print two integer numbers separated by a space — indexes of the first and the last volumes from each of the required time periods of Berlbury's creative work.

input
3 3 14 12 10
output
2 2 1 2 2 3

input
2 0 10 10
output
2 1 1 2

input
4 5 8 19 10 13
output
2 1 3 4