

Problems

Problem	MREŽA	SLALOM	A1
Source file	mreza.pas mreza.c mreza.cpp	slalom.pas slalom.c slalom.cpp	a1.pas a1.c a1.cpp
Input file	mreza.in	slalom.in	a1.in
Output file	mreza.out	slalom.out	a1.out
Time limit per test	5 seconds	5 seconds	5 seconds
Number of tests	10	10	10
Points per test	8	10	12
Total points	80	100	120
	300		

MREŽA

Two whole numbers **A** and **B** and a coordinate system in a plane are given. The origin **(0,0)** is labeled “**start**” and a point with coordinates **(A,B)** is labeled “**end**”. A figure is placed at the origin. A sequence of steps the figure has to perform is also given. Each step consists of a move from a current point to any of the four closest points with integer coordinates lying up, down, left and right. For example, moving one step up means going from (x,y) to $(x,y+1)$; moving one step left means going from (x,y) to $(x-1,y)$. If the figure is located at the point (x,y) then we calculate its *distance* from the end point by the formula

$$d((x,y), (A,B)) = \text{abs}(x-A) + \text{abs}(y-B).$$

Your task is to write a program that will find and remove a consecutive subsequence out of the given sequence of steps under the following two conditions:

- The distance between the end point and the last position of the figure is minimal possible.
- The figure never leaves the rectangle whose diagonal points are the origin and the end point and whose sides are parallel with the coordinate axes.

Input data

The first line of input file contains two whole numbers A and B, the coordinates of end point separated by a space character, $1 \leq A, B \leq 4000$.

The second line contains a whole number N, the number of steps, $1 \leq N \leq 8000$.

The third line contains a sequence of length N of characters. Each character of a sequence can be any of characters 'R' - right, 'U' - up, 'L' - left, 'D' - down. The sequence defines the sequence of steps the figure has to perform.

Note: Each set of input data will require removal of at least one step.

Output data

The first and only line of output file should contain two whole numbers K and L separated by one space character, $1 \leq K \leq L \leq N$, meaning that all steps beginning with K^{th} and ending with L^{th} should be removed from the sequence of steps given in the input file.

Examples

mreza.in

2 3
5
RURRR

mreza.out

3 4

mreza.in

2 2
5
RDLUR

mreza.out

2 3

mreza.in

3 3
11
UUUURRRRDDL

mreza.out

4 5

SLALOM

Janica did the first slalom rather bad and her only chance of winning the game is to do the second slalom as fast as humanly possible using the **shortest** route.

The beginning and the end of a path are two points in the plane coordinate system and the gates through which each skier has to pass are line segments **parallel with the x coordinate axis**. A sequence of numbers determining y coordinate of the start point, y coordinates of the gates given in the same order in which a skier have to pass them and the y coordinate of the end point is **strictly decreasing** sequence.

A path beginning with a start point and ending with an end point consists of a **sequence of line segments**. Two consecutive line segments share the same endpoint that lies on a line segment representing a gate. Each line segment in the sequence connects two consecutive gates. The start point is an end point of the first line segment un the sequence and the end point is an endpoint of the last line segment in the sequence.

Write a program that will determine **the length of the shortest path** connecting the star point and the end point.

Input data

The first line of input file contains an integer N , $1 \leq N \leq 5000$, the number of gates.

The second line contains two whole numbers X_S and Y_S separated by a space character, the coordinates of the start point.

Each of the following N consecutive lines contains three whole numbers X_1 , X_2 and Y separated by a space character, $X_1 < X_2$, meaning that end points of corresponding gate are (X_1, Y) and (X_2, Y) . The gates are given in the same order a skier have to pass through them.

The last line contains two whole numbers X_C and Y_C separated by a space character, the coordinates of the end point.

The absolute values of all coordinates contained in the input files will be less than or equal to 15000.

SLALOM

Output data

The first and only line of output file should contain the length of the shortest path, rounded to **four** decimal places, i.e., your answer should not differ from the correct answer by more than 0.0001.

Examples

slalom.in

```
2
4 4
2 4 3
3 7 2
6 1
```

slalom.out

```
3.8284
```

slalom.in

```
3
5 4
2 4 2
5 7 1
2 4 0
5 -2
```

slalom.out

```
7.3006
```

slalom.in

```
5
0 10
-4 -2 9
-8 -4 8
-6 -3 6
-8 -6 4
-10 -9 3
0 1
```

slalom.out

```
20.7627
```

A1

Our country needs to be connected by highways that need to be repaired every once in a while. A committee chosen by the government of our country tries to continually improve the condition of the main highway imaginatively named “**A1**” by finding the first damage that needs to be fixed. Starting with the first damage the highway is divided into segments of the same length. A damage fixing team is then being sent to each segment containing damage. Since there usually are much more damages than teams available, it would be advisable to **divide** the highway into segments of given length in such a way that the number of segments containing damages is **as small as possible**.

There are N known damages on the highway, each described by an integer representing its distance from the beginning of the highway in meters. The length of a segment is M meters (an integer). There are no damages **within the first M meters**. The first segment has to start within the first M meters of the highway. If the first segment starts at the K^{th} meter then L^{th} segment starts at the $(K+(L-1)*M)^{\text{th}}$ meter of the highway. A team can fix all damages occurring within one segment starting with the first meter of the segment and ending with the last meter of that segment.

Write a program that will determine the minimal number of damage fixing teams needed to fix all damages occurring on the highway “**A1**” and all possible beginnings of the first segment.

Input data

The first line of the input file contains two whole numbers M and N separated by a space character, $1 \leq M, N \leq 100\,000$, where M is a length of a segment and N is a number of damages occurring on the highway.

The second line contains N whole numbers separated by a space character representing the positions of all damages. Each of those numbers is greater than M and less than or equal to $2\,000\,000\,000$ (two billion). The sequence of N numbers given in the second line will always be strictly increasing sequence.

Output data

The first line of the output file should contain the minimal number of damage fixing teams.

The second line should contain all the positions where the first segment can begin. These numbers should be separated by a space character and they must form a **strictly increasing** sequence.

Examples

a1.in

```
3 5
4 5 7 8 9
```

a1.out

```
2
1
```

a1.in

```
4 3
7 14 15
```

a1.out

```
2
1 2 4
```

a1.in

```
2 10
3 4 7 8 12 13 14 15 20 21
```

a1.out

```
7
1 2
```