COCI '16 Contest 1 #2 Jetpack

Time limit: 0.6s   **Memory limit:** 64M

Little Mirko got a new mobile phone for his birthday! As all kids nowadays, he quickly downloaded all of the popular mobile games, including Jetpack Joyride.

In the game, the protagonist Barry is running across a field consisting of 10 rows and N columns of squares of equal size. Initially, Barry is located in the center of the square in the lower left corner. Barry is constantly running to the right at the speed of one square per second. Additionally, he must avoid obstacles that are in his way.

When Mirko presses the phone screen, Barry turns on his super-duper special jetpack and starts his ascent at the speed of one square per second (still moving to the right, now moving diagonally up at an angle of $45°$, until he reaches the ceiling, when he will continue moving to the right until Mirko releases the screen). When Mirko releases the phone screen, Barry starts falling down at the speed of one square per second (now moving diagonally again, but this time facing down, until he reaches the floor, when he will continue moving to the right).

Mirko just started playing the game recently and he's still not good at it. He saw on YouTube that someone managed to complete the game by crossing all N columns, so he is asking you for your help. He will give you the layout of the fields in the game, and you must output the moves he has to play in order to win.

Input Specification

The first line of input contains the integer N ($1 \leq N \leq 10^5$), the size of the field.
Each of the following 10 lines contains N characters $\boxed{.}$ and $\boxed{X}$, the layout of the field in the game. The characters $\boxed{X}$ denote obstacles, and $\boxed{.}$ walkable fields.

Output Specification

The first line of output must contain the integer P ($0 \leq P \leq 5 \times 10^4$), the number of moves Mirko has to make.
In the following P lines, output any series of P moves, each in its own line, such that it solves Mirko's problem from the task.
A move is determined by two integers t_i and x_i, where t_i denotes the second in which Mirko has to press the screen, and x_i denotes how long he needs to keep the screen pressed.

A series of moves must be sorted in chronological order. In other words, it must hold $t_i + x_i \leq t_{i+1}$.
Also, no move should begin after the end of the game, $t_i < N$.

The input data will be such that a solution will surely exist.

Sample Input 1

```
11
.....XX...X
....XX...XX
...XX...XX.
..........
....XXX....
..........
.....X.....
....XX...X.
...XX...XX.
...X...XX..
```

Sample Output 1

```
2
1 4
7 2
```

Explanation for Sample Output 1

The path Mirko has to take is denoted with ⟨ * ⟩ :

```
.....XX...X
....XX...XX
...XX...XX.
..........
....XXX....
.....*...*.
....*X*.*.*
...*XX.*.X.
..*XX...XX.
**.X...XX..
```

Sample Input 2

```
20
X..................X
.X................X.
..X..............X..
...X............X...
....X..........X....
.....X........X.....
......X......X......
.......X....X.......
........X..X........
.........XX.........
```

Sample Output 2

```
1
8 10
```