

Problem 3. Bombs

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

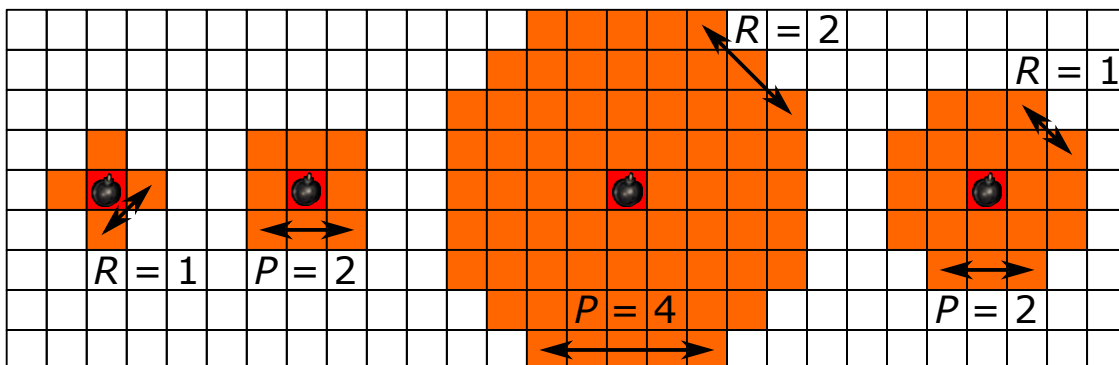


Gennadiy is playing his favorite game.

The game level is a field of square cells. Each cell is either empty or filled with ground. An empty cell can contain the level entrance or exit. The player can walk on empty cells, moving to a cell adjacent by side with each move. The player cannot move outside the field.

Some levels cannot be passed by regular means, and in such cases, bombs are used. A bomb can be stuck to the ground, and it will explode, clearing the nearby cells from ground. For the purpose of this problem, assume that the player can place a bomb in the cell where he is currently located if it is adjacent to a ground cell. After the bomb explodes, all cells in the explosion zone become empty and thus traversable. The player, level entrance, and level exit can safely be in the explosion zone, they are not affected by the explosion.

The explosion zone is an octahedron with the center in the cell where the bomb was placed. The explosion zone has two parameters P and R , and its precise shape is defined in the following manner. To traverse its border cells along the perimeter, you must move up P times then move up and right R times, then move right P times, move right and down R times, then move down P times, etc. until you return to the starting cell. The shape of the explosion zone with some values of the parameters is shown in the figure:



Bombs are a limited and valuable resource and thus should be used sparingly. Help Gennadiy pass a level with a minimum number of bombs.

Input

The first line of the input file contains two integers N and M — the vertical and horizontal size of the field, respectively ($1 \leq N, M \leq 1500$). The second line contains two integers P and R — the parameters of the explosion zone ($0 \leq P, R \leq 1000$). It is guaranteed that the number P is even, and that $P + R > 0$.

Next, the level map is provided, as N lines with M symbols in each. Each symbol describes the contents of the corresponding cell:

- @ — cell with ground.
- . — empty cell.
- S — empty cell with the level entrance.
- E — empty cell with the level exit.

It is guaranteed that the map only contains one entrance cell and one exit cell.

Output

In the first line of the output file, print an integer B — the minimum sufficient number of bombs ($B \geq 0$). In the remaining B lines, print the cells where bombs must be placed, in the order of their detonation. In each line, print two integers u and v — the indices of row and column, respectively ($1 \leq u \leq N$, $1 \leq v \leq M$).

Examples

input.txt	output.txt
<pre> 9 10 0 1 @@.....@@ @@S@@@@... @@@@@...@. @@@@@@@@@@@ @@@@@@@@@@@@ @@@.@@@@.. @...@@@E@. @@@@@@@@@@@ </pre>	<pre> 3 3 7 4 7 8 10 </pre>
<pre> 4 5 2 1 S@@@@ ..@@@ @@@E @@@@@ </pre>	<pre> 1 2 2 </pre>
<pre> 7 5 4 0 @E@@@ @@.@@ @@@@@ @@@@@ @@... @@... @@..S </pre>	<pre> 2 5 5 2 3 </pre>

Example explanation

In the first example, the explosion is cross-shaped and affects the cells sharing a side with the bomb cell. Two bombs are needed to break the wall in the lines 4 and 5, and another bomb to reveal the exit. In the second example, the explosion is powerful enough to create a passage with just one bomb. In the third example, two bombs are required. Note that it is forbidden to place the first bomb in cell (6, 4), because this cell does not contact ground.