

Задача Н. Звёзды за кормой...

Имя входного файла:	seastars.in
Имя выходного файла:	seastars.out
Ограничение по времени:	2 секунды
Ограничение по памяти:	64 мегабайта

В последнее время Дятел очень заинтересовался одной игрой. Вкратце правила таковы:

Имеется море размером $(2 \cdot n + 1) \times (2 \cdot n + 1)$ клеток. Изначально в некоторых клетках находятся острова, в некоторых — вражеские корабли, в центре стоит корабль игрока. За один ход игрок может либо переместиться, либо выстрелить; пропустить ход нельзя.

Перемещение производится на одну из восьми соседних клеток, при условии, что она существует и свободна. При этом считается, что корабль сначала поворачивается на месте носом к выбранной клетке, а затем двигается в неё.

При выстреле корабль производит залп обоими бортами. Ядра при этом летят перпендикулярно текущему курсу корабля. Ядра каждого борта поражают первую встретившуюся в данном направлении цель (корабль или остров), но не далее трёх клеток от корабля игрока (не считая клетку, на которой находится сам корабль игрока). Например, если корабль приплыл из клетки $(2, 1)$ в клетку $(1, 2)$, а затем выстрелил, то он уничтожит вражеский корабль на клетке $(4, 5)$, но не поразит корабль на $(5, 6)$. Если к тому же на клетке $(2, 3)$ будет стоять корабль или остров, то корабль на $(4, 5)$ останется цел.

Изначально корабль игрока стоит на клетке (n, n) с таким направлением, будто он приплыл из клетки $(n - 1, n)$.

После каждого хода игрока вражеские корабли одновременно делают ход. Каждый из кораблей ходит на ту из восьми соседних клеток, сумма модулей разностей координат которой и координат корабля игрока минимальна. Формально, если корабль игрока находится на клетке (x_s, y_s) , то выбирается такая клетка (x, y) , для которой $|x_s - x| + |y_s - y|$ минимально. Если при этом оказывается, что в данной клетке находится остров, то вражеский корабль погибает. Если хотя бы один вражеский корабль попадает на клетку с кораблем игрока, то игрок проигрывает. Если хотя бы два вражеских корабля оказываются на одной клетке, то они погибают и на этой клетке образуются обломки, которые далее действуют так же, как остров, за тем исключением, что снаряды перелетают через обломки, а не поражают их. При уничтожении корабля ядром тоже образуются обломки. На месте острова обломки не образуются.

Игрок побеждает, если на поле нет ни одного живого вражеского корабля.

Дятлу стало очень интересно, можно ли в каждой конкретной ситуации выиграть или нет. Для этого он решил написать программу. Нет, вам не надо ему помогать. Он с этой задачей уже справился, а вот справитесь ли вы?

Формат входного файла

В первой строке входного файла задано число n ($1 \leq n \leq 6$), определяющее размер поля. Далее следуют $2 \cdot n + 1$ строк по $2 \cdot n + 1$ символов в каждой. При этом i -й символ $(j + 1)$ -й строки описывает клетку поля с координатами $(i - 1, j - 1)$ (клетки нумеруются с 0). Значение символов следующее:

- ‘.’ (точка) — пустая клетка.
- ‘t’ (t маленькое латинское) — вражеский корабль.
- ‘O’ (o большое латинское) — остров.
- ‘+’ (плюс) — корабль игрока.

Символ ‘+’ всегда присутствует и располагается только в клетке с координатами (n, n) . Количество остальных символов может быть любым и ограничено только размерами доски.

Формат выходного файла

В первой строке файла выведите количество ходов, необходимых для выигрыша. В последующих строках выведите координаты корабля после каждого из ходов, по одной паре в строке. Минимизи-

рывать число ходов не обязательно; тем не менее, если вражеских кораблей уже не осталось, дальнейшие ходы делать не следует. Если решения не существует, выведите 'IMPOSSIBLE' в единственной строке файла. Если существует несколько решений, выведите любое.

Примеры

seastars.in	seastars.out
<pre> 5tt... 0.....t... t.....t0 ..t..+....0.....0t ...0..... ...t..... ..t..0.... </pre>	<pre> IMPOSSIBLE </pre>
<pre> 5 ...t..0.0. ...t.....0..+....t.....0.... </pre>	<pre> 7 5 5 5 5 5 5 5 4 5 3 6 2 5 1 </pre>