あーだこーだー特別配信 創立11周年記念
企業対抗 Team Battle

問題

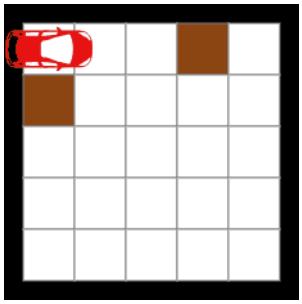
A - Turn Right

実行時間制限: 2 sec / メモリ制限: 1024 MB

ストーリー

AtCoder社の高橋社長はオフィスで車のおもちゃを走らせて遊んでいる。 このおもちゃは障害物にぶつかるまで真っ直ぐ動き、障害物にぶつかると右に90度向きを変える、という動作を延々と繰り返す。 この動作を繰り返していると、そのうち既に通ったことのある場所を同じ向きで通過し、そこから先は同じルートをぐるぐると無限ループすることになる。

高橋社長は車は大好きだが無限ループは大嫌いなため、オフィスに障害物を設置することで、無限ループするまでに出来るだけ長い距離を移動出来るようなコースを作成することにした。 高橋社長の手伝いをして欲しい。



無限ループするまでの移動例
(無限ループに突入するまでの移動距離18)

問題文

$n \times n$ マスの盤面がある。 一番左上のマスの座標を $(0, 0)$ とし、そこから下方向に i マス、右方向に j マス進んだ先のマスの座標を (i, j) とする。 初期状態で (si, sj) の位置に右向きの状態で車のおもちゃが置かれている。

いくつかのマスには初期状態で障害物が設置されており、これらの障害物は取り除いたり他のマスに移動させたりすることは出来ない。 また、$n \times n$ の範囲外のマスは全て障害物が置かれているものとして扱う。 残りの空きマスに好きなように障害物を設置せよ。 ただし、車のおもちゃの初期位置には障害物は設置されていないことが保証されており、新たに障害物を設置することも出来ない。

障害物の設置が完了すると、以下のルールに従って車のおもちゃが移動する。

1. 現在の向きに1マス進んだ先に障害物がある場合、その場で右方向に90度回転する。
2. 現在の向きに1マス進んだ先に障害物がない場合、その方向に1マス進む。

この動作を繰り返していると、そのうち既に通ったことのあるマスを同じ向きで通過し、そこから先は同じルートを無限ループすることになる。 無限ループに突入するまでの移動距離が出来るだけ長くなるように障害物を設置せよ。 ここで「無限ループに突入するまでの移動距離」は、以前と同じ位置かつ同じ向きの状態に初めてなった時点での、それまでのルール2の適用回数(最後がルール2の場合はそれを含む)を表す。

得点

無限ループに突入するまでの移動距離を L、 初期状態における空きマスの総数を E としたとき、 以下のスコアが得られる。

$$\mathrm{round}\left(10^6 \times \frac{L}{4E}\right)$$

出来るだけ高得点が得られるように障害物を設置するプログラムを作成せよ。

テストケースは全部で 100 個あり、各テストケースの得点の合計が提出の得点となる。 テストケースのうち $i = 1, 2, \cdots, 100$ 番目の入力は $n = 20 + \mathrm{floor}(31 \times (i - 1)/100)$ を満たす。 一つ以上のテストケースで不正な出力や制限時間超過をした場合、提出全体の判定が WA や TLE となる。 コンテスト時間中に得た最高得点で最終順位が決定され、コンテスト終了後のシステムテストは行われない。 同じ得点を複数の参加者が得た場合、提出時刻に関わらず同じ順位となる。

入力

入力は以下の形式で標準入力から与えられる。

$$
\begin{aligned}
&n \\
&si \quad sj \\
&b_0 \\
&\vdots \\
&b_{n-1}
\end{aligned}
$$

n は盤面の大きさを表す、20 以上 50 以下の整数値である。(si, sj) は車のおもちゃの初期位置を表す 0 以上 $n-1$ 以下の整数値である。各 b_i は `#` と `.` からなる長さが n の文字列であり、その $j(0 \leq j \leq n-1)$ 文字目は、座標 (i, j) のマスに障害物が置かれている場合は `#`、置かれていない場合は `.` である。

出力

新たに設置する障害物の総数を m とし、設置する障害物の座標を $(i_0, j_0), \cdots, (i_{m-1}, j_{m-1})$ としたとき、以下の形式で標準出力に出力せよ。

$$
\begin{aligned}
&m \\
&i_0 \quad j_0 \\
&\vdots \\
&i_{m-1} \quad j_{m-1}
\end{aligned}
$$

同じマスに複数の障害物を設置することは出来ず、車のおもちゃの初期位置のマスにも障害物は設置することは出来ない。 これらの条件を満たさない場合、WA と判定される。

例を見る (https://img.atcoder.jp/atcoder11live/f62fc84.html?lang=ja&seed=0&output=31%0D%0A19+19%0D%0A18+7%0D%0A2+9%0D%0A16+0%0D%0A17+18%0D%0A19+9%0D%0A0+12%0D%0A16+11%0D%0A1

入力生成方法

▶ 詳細

ツール(入力ジェネレータ・ビジュアライザ)

- Web版 (https://img.atcoder.jp/atcoder11live/f62fc84.html?lang=ja): ローカル版より高性能でアニメーション表示と手動プレイが可能です。
- ローカル版 (https://img.atcoder.jp/atcoder11live/f62fc84.zip): 使用するにはRust言語 (https://www.rust-lang.org/ja)のコンパイル環境をご用意下さい。
  - Windows用のコンパイル済みバイナリ (https://img.atcoder.jp/atcoder11live/f62fc84_windows.zip): Rust言語の環境構築が面倒な方は代わりにこちらをご利用下さい。

入力例 1

```
20
10 8
....................
........#...........
....................
....................
....................
.....#.............#
....................
...................#
```

出力例 1

```
31
19 19
18 7
2 9
16 0
17 18
19 9
0 12
16 11
13 0
```