Challenge: “Grid Golf”
Level: Non-beginner

Description:

Imagine a grid laid out on a wall in front of you, with each grid location having a number written on it. Putting your finger on the grid, you can move in any direction the number of spaces written on the location beneath your finger – as long as it doesn't send you off the edge or to an invalid position! One space that does not have a number written on it, but a letter "e" -- this is the "end" location, which is the goal of your movement around the grid.

Example input set:

A string representing a grid of numbers such that:

- Each grid location is indicated by one of the following:
    - An **s** followed by a number, indicating the location that movement will start from and the number of spaces to move when leaving that location; the **s** location will always be immediately followed by this number, which will always be 0-9
    - An **e**, indicating the location that movement should end on
    - A number, indicating a valid location to land on and continue movement from and the number of spaces to move when leaving that location
    - An X, indicating an invalid location; movement may pass over this location but may not stop on this location
- Each row ends with a space
- The bottom row of the grid is listed first, and the top row is listed last

```
21e2 1123 X211 s2X2X
```

This grid would be translated visually to:

```
s2 X 2 X
X  2 1 1
1  1 2 3
2  1 e 2
```

Sample solution:

Movement <u>always</u> starts from the location marked by the s. You are required to move 2 spaces when leaving this location.

```
s2 X 2 X
X  2 1 1
1  1 2 3
2  1 e 2
```

D – move down 2 spaces, landing on a space with a 1. This move travels over an X but does not land on that location, so it is a valid move. You are required to move 1 space when leaving this location.

```
s2 X 2 X
X  2 1 1
1  1 2 3
2  1 e 2
```

R – move right 1 space, landing on a space with a 1. You are required to move 1 space when leaving this location.

```
s2 X 2 X
X  2 1 1
1  1 2 3
2  1 e 2
```

D – move down 1 space, landing on a space with a 1. You are required to move 1 space when leaving this location.

```
s2 X 2 X
X  2 1 1
1  1 2 3
2  1 e 2
```

R – move right 1 space, landing on the space with the e.

```
s2 X 2 X
X  2 1 1
1  1 2 3
2  1 e 2
```

PASS/FAIL

The solution will be considered PASSED if it returns an output which:

- **Is properly formatted**
  Properly formatted steps will consist of a direction token, which will indicate the desired direction to move (the number of spaces to move will be determined from the number in the current grid location). Direction tokens are as follows:
  - U– indicates a movement "up"
  - D – indicates a movement "down"
  - L – indicates a movement "left"
  - R – indicates a movement "right"

  A path is a sequence of direction tokens with no spaces between them. For example:
  UURDLD

  If multiple paths are found, they should be submitted with a space between them. For example:
  UURDLD URRDDRDL

  (The submission of multiple valid paths will be used in tiebreaker scenarios. See the Tiebreakers section below for more information. If multiple paths are submitted, all paths must be valid.)
- **Contains only valid moves**
  Valid moves are moves with do not leave the grid defined in the input. Any move which would go beyond these bounds “falls off” the grid, and is invalid.
- **Does not land any square on the grid more than once**
  Movement may *pass over* a square on the grid any number of times, but movement may only *end* on any square a maximum of one time per path.
  Ends at the specified end location
- **Is unique**
  Multiple identical solutions for a single input set will not be considered valid.

Any solution which returns output that does not meet these criteria be considered FAILED.

To help visualize these movements mentally, imagine that the grid is laid out on a wall in front of you, placing your finger at the top left of the grid (Row 0, Column 0), and making movements from there. No turns/rotations are ever taken, your finger is only moved Up, Down, Left, or Right along a row/column of the grid.

Tiebreakers:

The code you submit will be called several times, with a different input set each time. If more than one team outputs a correct solution to at least one input set, the following tiebreaker scenarios will be used to determine the ordering of teams for this challenge, in the order given below:

- Most input sets solved successfully, highest number winning the tiebreaker. Each input set will be weighted equally for this tiebreaker (e.g. no extra points for solving a "more difficult" input set).
- Total number of valid paths found, summed across all input sets, highest number winning the tiebreaker. If Team A finds one valid path for input set 1, two valid paths for input set 2, and two valid paths for input set 3, their total number of valid paths is 5. Team B must find 6 or more total valid paths to win the tiebreaker.
- Submission time, earliest submission winning the tiebreaker.