2 Part : Path-finding in a grid 50%

In this exercise you will implement a class representing a terrain in the form of a rectangular grid. The cells of the grid are indexed with pairs of integers, where $(0, 0)$ represents the lower left cell of the grid, and $(m-1, n-1)$ represents the top right cell of the grid, where m is the width and n is the height of the grid. For example, a grid of width 4 and height 3 is indexed in the following way:

$$\begin{array}{cccc} (0,2) & (1,2) & (2,2) & (3,2) \\ (0,1) & (1,1) & (2,1) & (3,1) \\ (0,0) & (1,0) & (2,0) & (3,0) \end{array}$$

Some of the cells of the grid are *blocked by obstacles*. We are interested in determining whether there is a path to reach a given *goal cell* from a given *start cell* by moving one cell at a time in each of the four cardinal directions: **North, East, West, South**, and never entering a blocked cell or stepping outside of the grid. You will write functions that compute and output such a path if one exists, or otherwise report that there is none.

In the example above, suppose that cells (3,0), (3,2), (0,1) and (1,1) are blocked. We will represent grids in a textual form, denoting blocked cells with **x**, and denoting cells that are not blocked by **o**. Thus, the grid with these blocked cells is represented as:

o o o x
x x o o
o o o x

If the start cell is $(0, 0)$ and the goal is (0,2), then a path from start to goal is

$$(0,0), (1,0), (2,0), (2,1), (2,2), (1,2), (0,2).$$

We can represent this path in our textual representation as follows:

g * * **x**
x x * **o**
s * * **x**

Here, **s** denotes the start, **g** denotes the goal, and the * symbols denote the cells visited in this path from start to goal. Note that the letters **s** and **g** overwrite the * symbols from the corresponding cells of the path. If the start and the goal are the same cell, then the cell should be marked with **g**.

Alternatively, we can represent this path as a sequence of directions as follows:

```
Move from (0,0) to (2,0).
Turn North.
Move from (2,0) to (2,2).
Turn West.
Move from (2,2) to (0,2).
You have arrived at your destination.
```

You will have to implement functions for reading the description of a grid from a given file and computing a path for the start to the goal location in the grid, or determining

that there is no such path. Recall that a path should not enter blocked cells or exit the grid. You will also have to implement functions for outputting a computed path, in each of the two formats above: as the textual representation on the grid, and as a sequence of move and turn directions. Finally, you also have to implement functions for getting and changing the state of the grid (blocked cells, and start and goal locations).

A detailed description of the functions of the class is given in `Grid.h`.