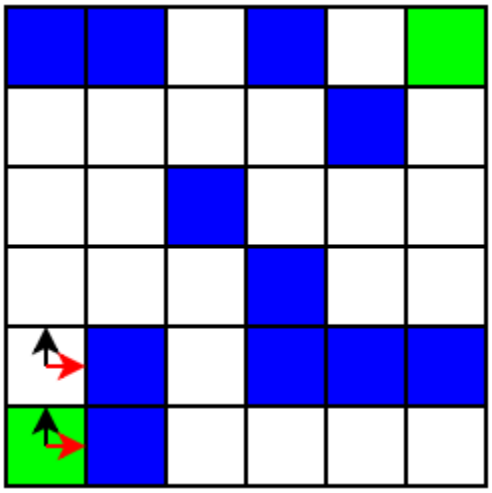
As discussed in the class, you are required to implement DFS, BFS, Greedy and UCS algorithm on a maze of any size with any combination of obstacles. A maze is provided below for your reference, where blue indicate obstacles and green cells indicate start and destination. The bottom left corner is the start location while top right is the goal in the following case.



For visual representation of maze, such as depicting the cells, agent location and chosen path, cells explored you can choose any scheme of your choice e.g., color coding, use of specific markers to represent obstacles, path or location.

Following are important points with respect to assessment of code:

1. Only running code will be marked, no partial marks for error prone code.
2. Implement the code in python.
3. The size of maze and location of maze should be generic.
4. Input about the maze can be taken from command line or from file.
5. The output for each algorithm must describe the actual path chosen, all of the cells explored while finding the optimal for ach algorithm.