Nov. 18/2022

Robots!

Robots! is a multi-round, completely autonomous game played by two robots taking turns moving across a 7x7 game board in search of a randomly placed prize token.

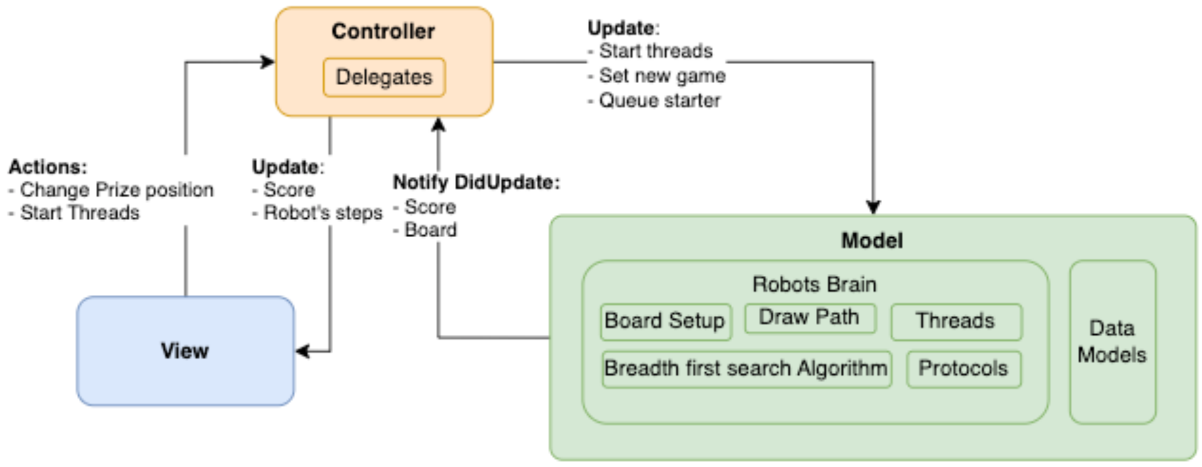
1. Used Stack and Concepts

- UIKit
- Swift
- Protocols and delegates
- MVC
- Threads and Grand Central Dispatch
- Graphs, nodes, and queues
- Breadth First Search Algorithm
- XCTest and UITest

2. Bonus

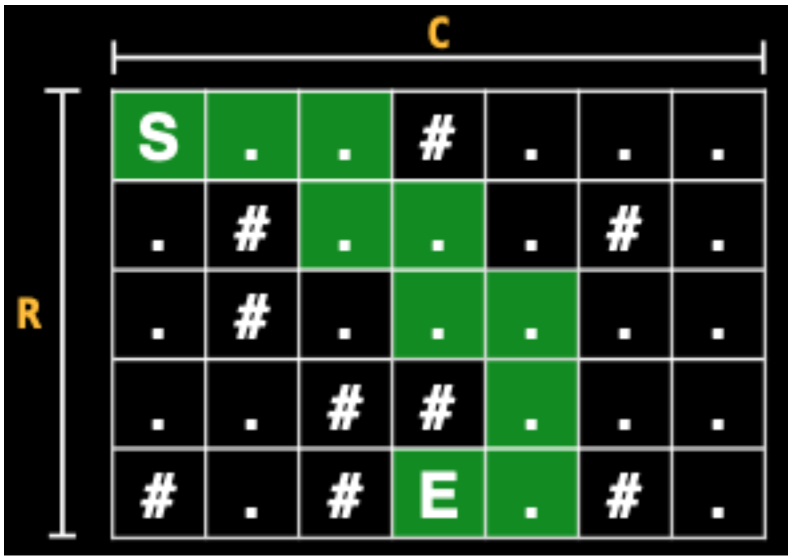
You can change the target by tapping one of the available nodes, so the robots change the path and try to reach the new goal.

3. Architecture

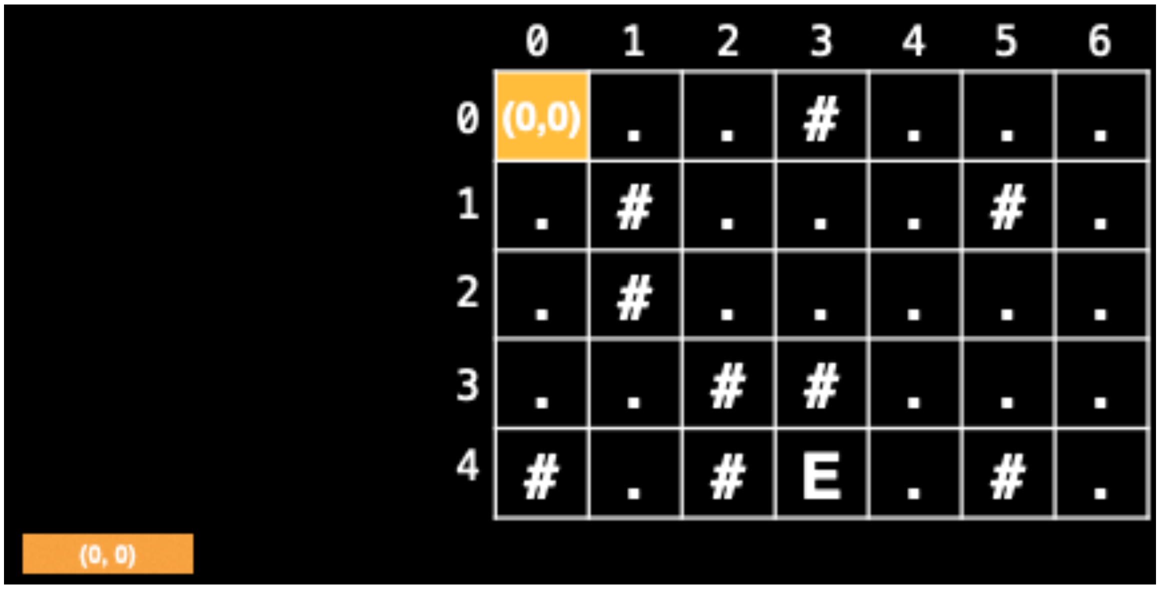


4. Breadth First Search Algorithm

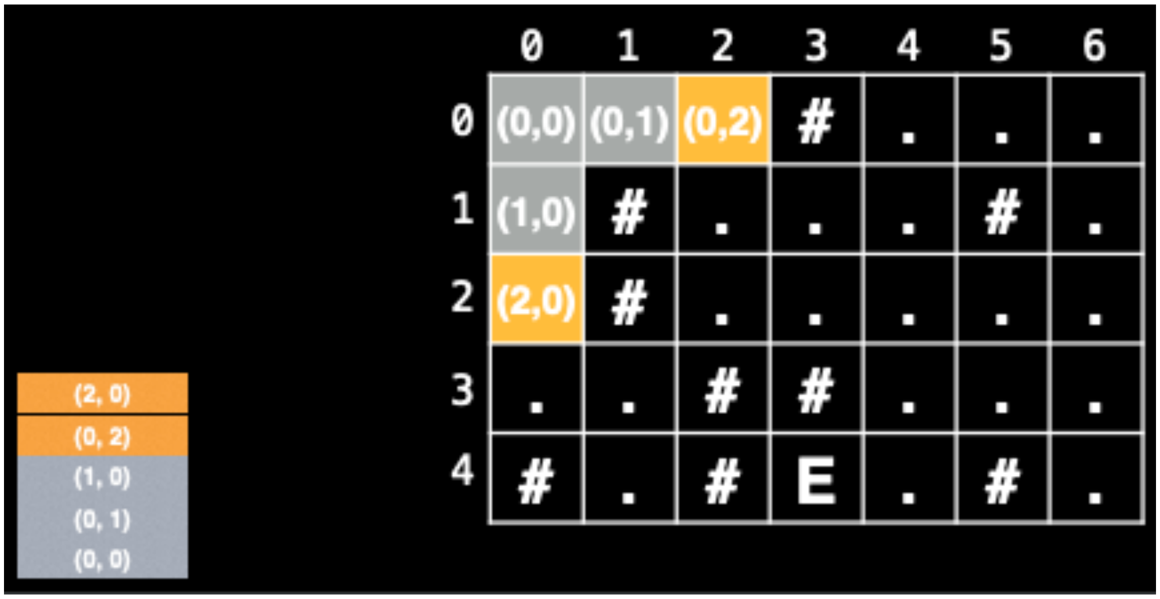
A dungeon has a board size of R x C and you start at cell '0,0' and there's an exit at cell '4,3'. Blockers are indicated by a '#' and empty cells are represented by a '.'.

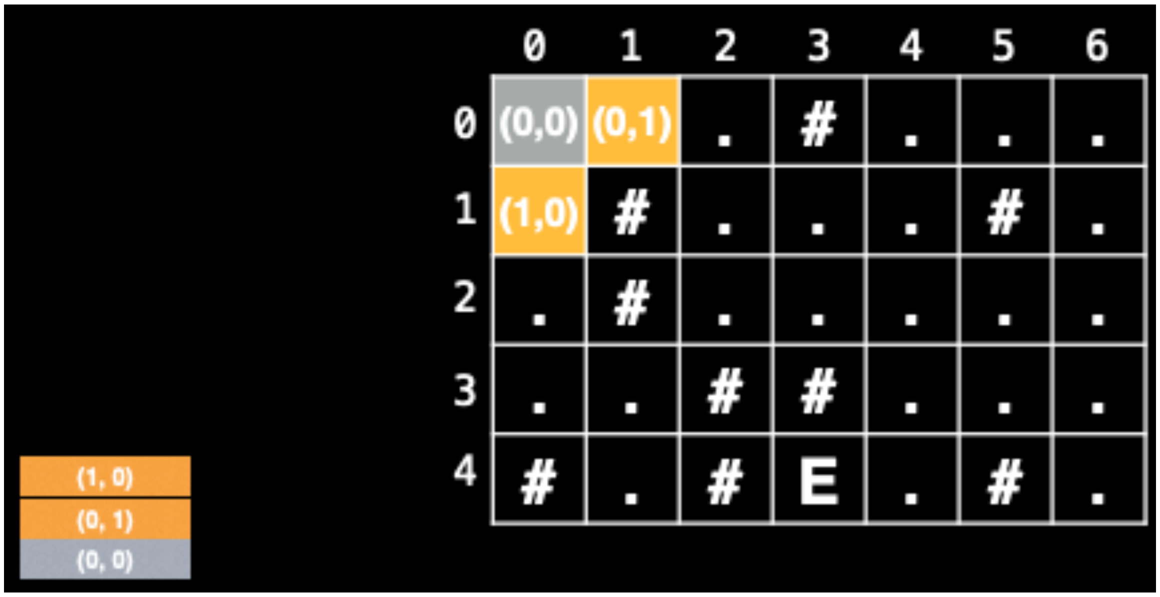


Start at the start node coordinate by adding (sr, sc) to the queue:

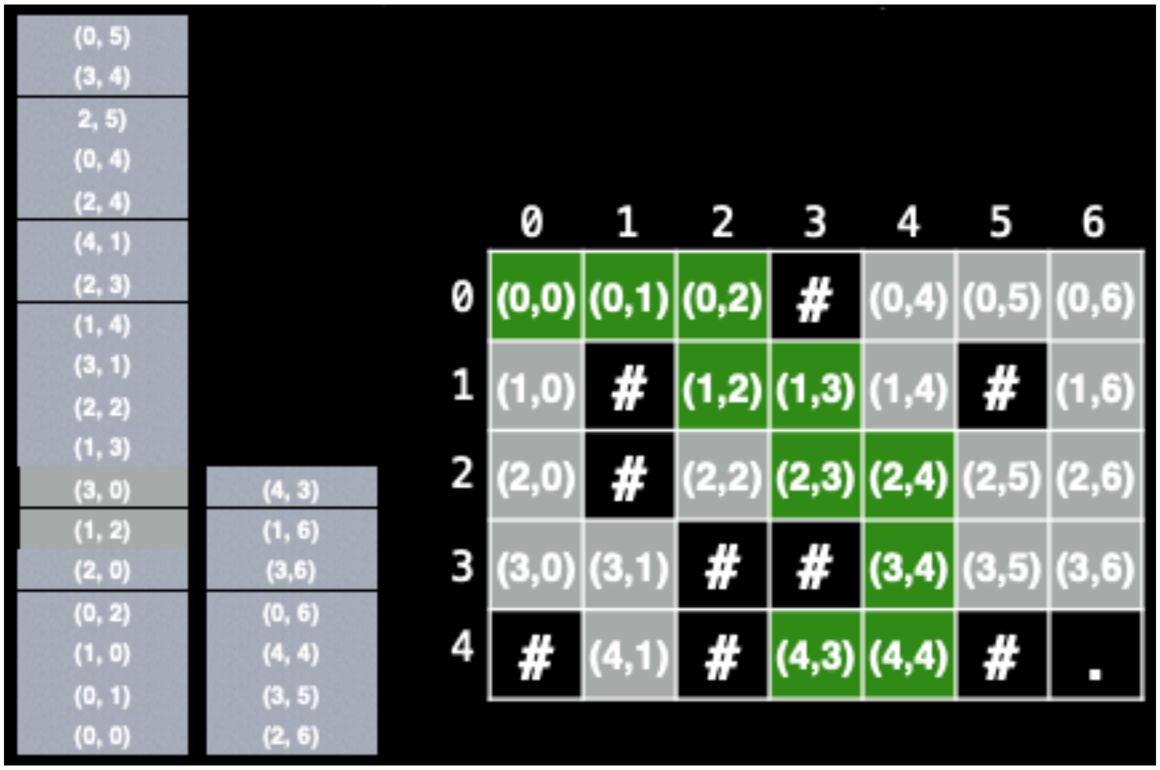


Keep adding to the queue:





We have reached the end, and if we had a 2D prev matrix we could regenerate the path by retracing our steps:


(0, 5)
(3, 4)
2, 5)
(0, 4)
(2, 4)
(4, 1)
(2, 3)
(1, 4)
(3, 1)
(2, 2)
(1, 3)
(3, 0)
(1, 2)
(2, 0)
(0, 2)
(1, 0)
(0, 1)
(0, 0)
(4, 3)
(1, 6)
(3,6)
(0, 6)
(4, 4)
(3, 5)
(2, 6)
0 1 2 3 4 5 6
0 (0,0) (0,1) (0,2) # (0,4) (0,5) (0,6)
1 (1,0) # (1,2) (1,3) (1,4) # (1,6)
2 (2,0) # (2,2) (2,3) (2,4) (2,5) (2,6)
3 (3,0) (3,1) # # (3,4) (3,5) (3,6)
4 # (4,1) # (4,3) (4,4) # .