Amazing Algorithms

Daniel Rinehart
http://danielr.neophi.com/
Boston Software Craftsmanship Aug 5, 2013

Overview

Simply connected maze, only one path between two points, no loops, or enclosed spaces.

Maze exit maybe anywhere, not necessarily on an outer wall.

Structured as if you were in the maze trying to find your way out.

Solver Loop

Framework creates solver and repeatedly calls:

next(currentRoom) -> nextLocation

currentRoom has x, y, z, and exits
each exit has x, y, z

nextLocation is x, y, z of one of the exits

Framework checks for finish

Random Mouse

Randomly pick an exit each time.

This trivial sample is implemented in each language as an example.

Wall Follower

Use either left-hand rule or right-hand rule.

Keep one hand in contact with one wall of the maze and pick the exit which follows that rule.

Trémaux's

A path is either unvisited, marked once, or marked twice.

Every time a direction is chosen it is marked by drawing a line on the floor (from junction to junction).

In the beginning a random direction is chosen (if there is more than one).

On arriving at a junction that has not been visited before (no other marks), pick a random direction (and mark the path).

When arriving at a marked junction and if your current path is marked only once then turn around and walk back (and mark the path a second time).

If this is not the case, pick the direction with the fewest marks (and mark it, as always).

Resources

http://en.wikipedia.org/wiki/Maze_solving_algorithm

http://www.astrolog.org/labyrnth/algrithm.htm#solve