

Worksheet 11

Lefty can navigate through a maze

Robotics Course Instructional Guide



This worksheet brings together a number of tasks covered in previous sections of the guide book.

The objective is to develop a program that enables the robot to navigate a simple maze using the left-hand wall-following technique.

Activity

Linked

Skill Level

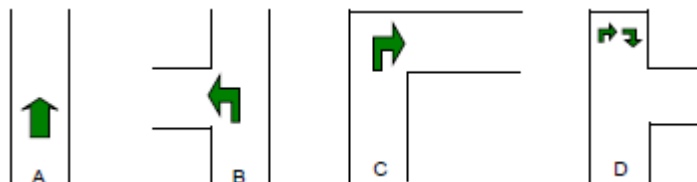
Intermediate

This exercise is probably one of the most challenging and rewarding things you can do with the robot. Making it move around and navigate a maze without it being touched appeals to most people regardless of their age. It also opens up the possibility of engaging in competitive challenges with friends to see who has the most agile and cleverest robot.

If you do some research on the Internet you will discover some of the many different methods people have developed for escaping from the centre of a maze or navigating through it. The most common method is the left-hand wall-following algorithm. An algorithm is a procedure or formula for solving a problem. It can be written informally as words or statements or in a structured way like a piece of pseudo code or a formal programming language.

Let's start off by assuming the robot has been placed at the entrance to a maze and see how a workable algorithm could be developed. At this stage it might be a good idea to share this exercise with a friend as you can bounce suggestions off of each other.

The robot has a set of distance sensors (positioned at the four corners and four edges) that can be used to detect if any obstacles are in the way. What the robot needs to do is check the left-hand distance sensor for the presence of the wall. It also needs to detect the distance sensor at the front of the robot to make sure the forward path is clear, before moving forwards. Situation (A) diagram below.



If the wall (on the left-hand side) disappears, indicating a gap in the maze, then the robot needs to rotate to the left and continue following the wall. Situation (B) in the diagram.

If the robot detects its forward path is blocked, as in situation (C), then it needs to stop and rotate clockwise by 90 degrees, and then reassess the situation. In this situation the robot can proceed forward.