

Take-Home Exam (Due October 3rd, during class time)

Introduction

Micro-mouse is an engineering design competition created by IEEE in the late 1970s. Small robotic "mice" solve a 16x16 cell maze. The mice are completely autonomous and must find their way from a predetermined starting position to the center of the maze. The mouse needs to keep track of where it is, discover walls as it explores, map out the maze and detect when it has reached the center. The mouse typically performs additional searches to determine the fastest route possible from the start of the maze to the center. Figure 1 (left) shows a typical maze used for a micro-mouse competition, and Figure 1 (right) shows a picture of the *Japan Micromouse Contest 2011*.

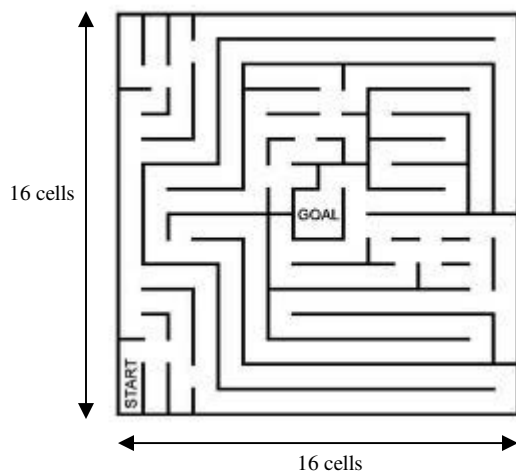


Figure 1 (left) a typical maze, (right) a picture of a recent micro-mouse competition.

Solving mazes has been studied for years and years in mathematics, which means there are too many algorithms to name here. The simplest maze solving algorithm is the random algorithm, which just goes forward in the maze until it comes to a wall. It then randomly chooses right or left and then goes forward again. With this algorithm, the mouse will always find the destination (if there is no time limit), but it will almost definitely use a very inefficient route to get there.

The flooding algorithm is based on quite an ingenious thought. Imagine you are stood in a maze which has non-permeable walls and flat, level floors. You also have a hosepipe. If you turn the hosepipe on, the water will start filling the maze. The shortest route to the target destination will be taken by the first drop of water that arrives there. This is why it is called the 'flooding' algorithm. This algorithm ensures you get the shortest route (not necessarily the fastest though). We will implement the flooding algorithm in this project since it is a fairly easy concept and produces good results.

Take-home requirements

In this exam you are required to:

- Implement the flooding algorithm in MATLAB, to be able to navigate through the maze shown in Fig. 1, from the START position, to the GOAL position
- The program should produce a map of the shortest path found by your algorithm.

Your algorithm has to obey the micro-mouse rules set forth by the IEEE (you can find more information at this link: <http://iee.ucsd.edu/files/micromouse-rules.pdf>). However, in your program there is not a danger for the robot to slip or get stuck, so the 10 minutes rule does not apply. Also assume that you do not have to correct for misalignments of the mouse

with respect to the walls, and the sensors in your ‘virtual’ micro-mouse can sense front and side walls in the north, east, and west directions.

Items to hand in (by email)

A word document which will contain the following items:

- Names of the team members. Include the name of the project (if you want to give it a name), and the date. All this information goes in the front page.
- In subsequent pages you need to include:
 - (10 %) A detailed description of the project, and your algorithm.
 - (20 %) A flow chart of your algorithm.
 - (70 %) A well documented MATLAB program. Use functions as much possible to avoid having a long program which will be hard to debug and follow. This improves the readability of the program
- Include any references in the last page of the document.

A hand-coded solution of the maze in MATLAB is unacceptable. You must use basic MATLAB statements only, and you are not allowed to use any built-in function that gives a “canned” solution to this specific problem.

Academic honesty

All teams are expected to demonstrate personal integrity. Teams are allowed to consult the internet, textbooks, reference books, and class notes. Exchange of information during this take-home examination is strictly prohibited. Team of students who consult resources other than the ones mentioned above, will be penalized heavily. Take-home exams turned in later than the due date will not be accepted.