

Lab 11

Thanksgiving AI Challenge

Z. Hutchinson
zachary.s.hutchinson@maine.edu

November 23, 2021

Instructions

This lab can be submitted any time between now and Sunday Nov 28th. This lab is designed to give you just a glimmer of how hard it is to program an AI to do anything slightly complicated. It isn't important that you solve the task in any specific way. Just play around. See what you can accomplish. And submit your lab11.py file when the lab is over.

There are two Python files on Brightspace: *lab11.py* and *graphics.py*. The graphics file is the same one used in project 1 and 2. I am providing it here so you don't have to hunt for it. The lab11 file is a program that simulates an artificial agent in a 2D environment. Your task is to write an AI for the agent.

You do not need to understand all the code in lab11. You only need to understand what is contained in the parameters passed into the *GetMove* function. At the top of the file is a block of comments. Read them now.

Below the comments is a mostly empty function: ***GetMove***. Your task is to add to this function code which, given the limited information provided to the function, will choose a move for the agent. To do this, you should examine the current state of the world as seen by the agent. The agent can only see 1 cell in all directions. This information is contained in the *cells* parameter. It knows where it is (x,y) via the *player* parameter. And it knows where the goal is (x,y) via the *goal* parameter.

I have defined all the valid moves in constants at the top of the file (e.g., N, W, E, NE, etc.). I suggest you use these. I have also placed all of them in list called DIRS...if that's helpful.

The values in the *cells* parameter are integers. The values mean the following:

- 0 Open space. Agent is free to move here.
- 1 Obstacle. Agent cannot move here.
- 2 Goal. Move here and you win.
- 8 Agent path. An 8 denotes an open space in which the agent has already been. Agents can move into these spaces.

At any time, you can press 'Escape' to quit the simulation.

The **goal** is to write an AI routine which efficiently finds the goal in an unknown environment. However, I do not expect anyone to accomplish this in the 50 minutes of the lab. If all you get is an outline of how you would solve the problem, that is fine. The point of this exercise is to practice coding and rub shoulders with the difficulty of artificial intelligence problem solving.

More Challenge?

If you want more of a challenge you can change some of the parameters of the task. Many of these are defined in *main()* near the bottom of the file. I suggest:

- Add more obstacles by increasing the *obstacle_percent*.
- Limit the agent to only NSEW moves/view.
- Increase the amount of the world the AI can see: *agent_sight*. HINT: This doesn't make things easier. Just more info to process.
- Increase the size of the map.
- Ignore the goal and player location. Have the agent build a map of the environment.