EECS 391: Introduction to AI (Spring 2012) Programming Assignment 1 (Max Points: 100)

Assigned Thursday January 19, due midnight Tuesday February 7. Turn in your code using blackboard. Please comment your code extensively so we can understand it, and use sensible variable names. If two people worked on a submission, please include both your names and IDs in a README.

In this assignment, you will (i) familiarize yourselves with SimpleRTS and (ii) implement the A* search algorithm for pathfinding in SimpleRTS. SimpleRTS is based on Warcraft, a real-time strategy (RTS) game (though SimpleRTS is not real-time). RTS games generally have "economic" and "battle" components. In the economic game, the goal is to collect different types of resources in the map. Typical resources are "Gold" and "Wood." Resources are collected using units called "Peasants." Having resources allows the player to build other buildings (Peasants can be used to build things) and produce more units. Some of these units are battle units that can be used to fight the opponent. Games generally end when one player has no more units left; however, in SimpleRTS, a variety of victory conditions can be declared through XML configuration files. For example we can declare a victory condition to be when a certain amount of Gold and Wood have been collected, some number of units of a certain type built, etc.

Familiarization with SimpleRTS (20 points)

Starting with the code for RCAgent, and the rc_3m5t map in the data directory, write a SimpleRTS agent that executes the following actions: (i) collect enough Gold to build two Peasants, (ii) build two Peasants, (iii) collect enough Gold and Wood using all three Peasants to build a Farm, (iv) build a Farm, (v) collect enough Gold and Wood using all three Peasants to build a Barracks, (vi) build a Barracks, (vii) collect enough Gold using all three Peasants to build two Footmen, (viii) build two Footmen. Call your agent RCAgent2. Check the execution of the sequence of actions for your agent using VisualAgent.

Pathfinding (80 points)

Write an agent that can move around a given map by implement the A* search algorithm discussed in class. Note that SimpleRTS has built in COMPOUNDMOVE/createCompoundMove() actions that can handle pathfinding; **do not** use these actions in your code! Instead, use PRIMITIVEMOVE/createPrimitiveMove()s and implement your own pathfinding solution. For A*, you will need a heuristic function. The Chebyshev distance is a good heuristic for this purpose. This is defined as follows: $D((x_1,y_1),(x_2,y_2))=max(|x_2-x_1|,|y_2-y_1|)$. To test your algorithm, use the maze maps provided. In each map, a Footman is trapped in a maze. Somewhere in the maze is an enemy Townhall. Use your implementation to guide the Footman to the Townhall and attack it. When the Townhall is destroyed, the game will end. If it is not possible to guide the Footman to the Townhall, print "No available path." in the terminal and quit. Use VisualAgent to check

that your agent is behaving correctly in all cases. (You can run the maze maps just using VisualAgent, left click on the Footman and right click on the Townhall to see the solution according to the built in pathfinding routines.) Note that we will test your code on other maps as well, so ensure nothing is specific to the maps provided.

Code and Data Structures

Call your main agent containing the A* search code SearchAgent.java. Remember to have SimpleRTS.jar in the classpath when you compile this. To implement A*, it is fine to use basic data structures, such as arrays and lists, to keep track of what you need. You need not optimize for space or runtime for any of the methods (of course, you are welcome to do so if you wish).

What to turn in

Prepare a ZIP file with your agent code (do not include any class files). Include a README with your name(s) and ID(s) and any other comments you have, such as special compilation instructions if any. Name your file as "yourname_PA1.zip" and use Blackboard to submit it.