COSC 370 – Artificial Intelligence
Project 3

Purpose: Use any or all of the AI techniques from this course to program an AI for a competition simulation.

Task: For this project you will be coding an AI to play the adversarial simulation created by MIT for their annual Battlecode competition. For this "game" you will need to develop an intelligent agent that handles both macro and micro decision-making in real-time. From the Battlecode website (battlecode.org):

"In Battlecode, two teams of virtual robots roam the screen managing resources and executing different offensive strategies against each other. Your AI player will need to strategically manage your robot army and control how your robots work together to defeat the enemy team."

For this project, we will be using the Battlecode 2020 competition "Soup". You will find the links required to install and run the Battlecode 2020 application on the course website. You should feel free to use any AI techniques that you wish, including ones we have not covered in this class.

Implementation: You will use Java, version 8, the language allowed by Battlecode's simulator to create your intelligent agent. Likely you are using a JDK that is newer than JDK 8, so you should make sure to install JDK 8 from the links on Battlecode's "Getting Started" page before you attempt to build the simulator environment. **You may need to adjust your JAVAHOME environment variable to point to JDK 8, especially on Windows** (https://confluence.atlassian.com/doc/setting-the-java_home-variable-in-windows-8895.html). I also had some issues running the build commands from Eclipse and had to run them directly from the command line (https://2020.battlecode.org/common-issues), your mileage may vary.

Once you have completed the steps in the "Getting Started" document, you should make sure you are testing your implementations by running them in the Battlecode 2020 simulation. As a note, a link for the RobotController API is available on the course website (read through the document and find the RobotController section). These are the functions that run your robots. I've also linked a YouTube playlist that may be helpful.

I would strongly suggest looking at all of the links on the left side of the Battlecode 2020 homepage. They are very helpful. You should also look at the specification link on the course webpage which includes packages that you are allowed (and disallowed) to use, as well as notes on how winners are determined.

You should test your implementations not just against the example AI (which you should be able to beat relatively quickly), but also AIs from other teams!

Installation/Execution: Follow the installation instructions posted to the website for your OS and IDE. Please be sure to do this and test your installation by running a test game by Thursday, March 11.

Installation Warning: make sure that you remove any special characters or spaces from your bot folders. If you don't, the application may not be able to run your bots!

Competition Tournament: During the final period, we will be doing a single-elimination tournament pairing teams in the simulator (will be using pre-selected maps), best-of-3. The results will not directly affect your grade for this project, but success in the tournament is an indicator of a good solution and implementation.

You are required to work in teams of 4 for this project, assigned after class on March 4th. Please designate one person as your "team leader," this is the point person that will be handling any communication with me and other teams. They will also be the one responsible for setting up and providing your AI for the competition tournament as noted above. Please also provide a team name. Team names with point person information will be posted in order to facilitate scrimmages.

You will also be asked to provide a rubric evaluation for each of your teammates (posted to the course website). This joint assessment, as well as GitHub contribution information (see below) will be 30% of each student's grade and will be kept confidential.

Presentation: During the final period, you and your group will present your strategy in a 5-minute elevator pitch. It should include what algorithms you used, and any pertinent details of how you tackled the problem. You have leeway of 10% on either side of the 5-minute time target, going under or over will result in a penalty. Incomplete discussions of your strategy will also result in a penalty.

Github Use: Github is required for storing and updating your code. Each member must do their own pull requests (you are not allowed to have a central person handling all changes). This is done for two reasons: to make sure everyone gets experience using a source control mechanism and to make sure that everyone is contributing equally. We will be dedicating a class to the ins and outs of github and git. Your group leader is responsible for setting up the initial repository and should add me as a collaborator.

Learning Targets: direct application of AI techniques.

DUE: Your robot's source is due April 29th, 11:59pm via Blackboard. Team evaluations due April 30th by noon via email. Presentations and tournament will happen during the final period on April 30th at 2pm. All team members are required to be present.

Individual Rubric:

30% - team evaluations

15% - presentation
55% - AI implementation

Deductions:
-10 team evaluations not in correct format/turned in on time
-25 lack of github activity
-15 lack of comments in the code
-100 AI implementation not able to be tested in battlecode