CS255 Artificial Intelligence
Coursework: Robocode robot tank

Introduction

Your task in this coursework is to design and implement a robot tank for the Robocode environment. "Robocode is a programming game where the goal is to code a robot battle tank to compete against other robots in a battle arena. [. . .] The player is the programmer of the robot, who will have no direct influence on the game. Instead, the player must write the AI of the robot telling it how to behave and react on events occurring in the battle arena. Battles are running in real-time and on-screen."[1]

The task

You should use appropriate AI techniques to design the control of your robot tank, and then implement your design using Robocode. You may use any AI techniques, but should justify your choices and, if you combine techniques, should explain how they interact.

There are two deliverables for this task: (i) a report describing your design and (ii) your robot implementation. Your robot will be evaluated using a range of battle field sizes, a selection of competitors, and will be run in both one-to-one and melee (a set of robots picked at random battling against each other, with no teams) competitions.

You **must** use the Java implementation of Robocode, version 1.9.2.4, **as supplied in the zip on the module webpage**[2]. Do not use any other version or one of the betas. Your code should only use the standard Robot API, i.e. you should extend the Robot class (and **not** the AdvancedRobot). You should only use standard Java, using version 1.7 as installed on joshua. Ensure your code compiles on joshua before submission. Non-compiling code will be docked marks. No external Java APIs are permitted without prior permission — if you wish to use an external API you **must** get permission from the module organiser first.

The report and submission

Reports should be of 2000-3000 words and typewritten in 11pt font. You should submit your robot (source files as detailed below) and your report online using Tabula. Your report should also be submitted to the cabinet in DCS.

Your source (.java) files should be submitted as a zip to Tabula. This zip should contain **only** your java files and supplementary resource files. The java files should be in the correct directory structure for their package name. The top-level package name should be of the form uXXXXXXX where XXXXXXX is replaced with your University ID Number. An example submission is available in the zip from the coursework webpage.

---

[1]http://robocode.sourceforge.net/docs/ReadMe.html
[2]http://www2.warwick.ac.uk/fac/sci/dcs/teaching/material/cs255/coursework/

Hints

- There is extensive documentation on Robocode, including the API and example robots, online at http://robocode.sourceforge.net.
- In developing your robot you should consider the key choices that must be made. The RoboWiki[3] gives example strategies for radar, movement, and targeting and you can use these as a starting point for your design.
- When you are developing your robot you might want to test against other robots. For example robots to test your robot against, see http://robocoderepository.com/
- You can use any AI techniques, but you might want to consider GAs, rule-based systems, and/or plan libraries.

Deadline

The deadline for submission of your report and robot is **noon 5 May 2015**.

Marking Scheme

30% Strategy design
30% Justification and description of design
20% Quality of implementation
20% Performance of robot

Cheating/Plagiarism

The report and robot that you submit is an **individual** piece of work. All submissions will be put through plagiarism detection software which compares against a number of sources, including other submissions for CS255, submissions at other universities, web sources, conference papers, journals and books. Your robot will be checked against the example robots that exist, and you **must build your own robot**. While the use of literature sources and discussions of general concepts/ideas is a healthy academic activity, any work submitted for assessment must not include that of others. Similarly, although you may discuss your design and implementation with others, your coursework submission should be undertaken individually. You should take care to reference other work appropriately. Please see the student handbook for more information, or ask if you need guidance.

---

[3]http://robowiki.net/