EECS 391 (Spring 2012) Programming Assignment 6 (Max Points: 50).

PLEASE READ CAREFULLY! This is an extra credit optional assignment. If your final grade in EECS 391 without this assignment evaluates to D or lower, and you have done this assignment, then the assignment grade will be added to your programming assignment total points and a new grade will be computed. If higher than the original, you will receive the revised grade, which will in all cases be at most one level higher than the original (i.e. a D can become at most a C, and an F can become a D). If your final grade in EECS 391 without this assignment evaluates to C or above, this assignment WILL HAVE NO EFFECT. It is OK for two people to collaborate on this assignment as usual. However only the person(s) getting D or lower without this will have this count, as above.

Assigned Tuesday April 17, due midnight Thursday May 3. 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 write an agent that integrates the previous scenarios you have written code for in SimpleRTS. Download the PA6.zip file from the website.

1. Problem Setup

The scenario you will solve is built around the "integration.map" map in the zip file and the defaultConfig configuration file. In this map, you start with a peasant and townhall. Some resources are available near your starting point. Most of the map is hidden initially. Other resources are available in the hidden areas; however, there are also several enemy units hidden there. Your goal is to collect resources, locate the enemy units, build an army and destroy them. To do this, your agent can build additional peasants. It can also build farms which provide food and allow you to build additional units. It can build barracks which produce footmen. Finally it can build footmen, which can battle the enemies.

You can use code from all your previous assignments to help solve this map. You have solved all the individual pieces in assignments through the semester. Now you have to figure out a framework to put everything together. One possibility is to have a high-level "strategic" planner that chooses from the actions of resource-collection, scouting, build-army and combat. Once a high-level action is chosen the appropriate action is delegated to an execution module that is basically the code you already have to solve that task. This high-level planner can simply be a fixed rule-based action chooser for this assignment (if state looks like X, do Y). Note that multiple such actions may be happening in parallel (some units may be scouting while others collect resources for example). You are free to think up alternative frameworks and implement them. In the README accompanying your submission, write a few paragraphs explain how your integration scheme works.

To help you we have provided the "integration_demo.map" which removes the fog of war so you can see where everything is. However, as before, do not tailor your code to this map. We will test your code with other maps.

5. 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 an explanation of your integration framework. Also include any other comments you have, such as special compilation instructions if any. Include both your names in the README if you worked as a pair for this assignment. Name your file as "yourname_PA6.zip" and use Blackboard to submit it. **This zip file should only contain your source code, readmes and makefiles, not executables/object files/data files/anything else, and must be timestamped by the due date to avoid a late penalty.**