

Program #3: RoboCup Teams

Your assignment is to design a team of agents that can play simulated soccer. We will use the simulator designed for the Robocup tournament to test your agents by pitting them against each other in a tournament during our scheduled final exam period. This will be a team project with four teams of three people each.

Important dates:

Form your team	Tuesday, Nov. 2
Beta demo	Thursday, Nov. 18
Tournament	<i>tentatively Friday, Dec. 10 (our scheduled exam time)</i>

Forming teams:

You must decide on who your teammates will be by class on Tuesday, Nov. 2. As I said before, each team should consist of three people. In forming your team, you should consider having at least one Unix expert and one person who has a good understanding of the agents concepts discussed in this class. It may also be advantageous to have a member who is familiar with the game of soccer. Please give me a sheet of paper with your team name and the names of your team members. If any of your team members need accounts on the department's Solaris machines, please mention that and provide their Lehigh user ids.

Getting started:

In order to do well on this project, you will have to spend a lot of time designing and programming, so get started right away. You should begin to read the manual for the RoboCup soccer server as soon as possible and figure out how to use it. The course pages has an FAQ that you may find useful. Also, we have ported the server to work with the department's Solaris machines and have place these binaries `/export/robosoccer/` directory of `robosoccer.cse.lehigh.edu` (actually, this machine is `europa.cse.lehigh.edu`). We are using version 9.4.5 of the soccer base (the latest official release). You may use this machine to run your soccer server, however, note if other members in the class are using the server at the same time, you may have to specify a different port number for the server (as well as your monitor and clients) to use (the default port number is 6000). You are also welcome to copy these binaries and run them on different machines.

In order to allow you to spend more time on designing your agents, I encourage everyone to use one of the libraries available from the RoboCup library page (a link is provided from our course page). However, it is likely to take some work to get the C/C++ libraries compiled for Solaris. Therefore, I am offering *extra credit* to the first team that gets any of these libraries working and then makes available the final source and binaries for use to the other teams. Note, there is also a Java library (called Atan), which should be much easier to use directly, but if you choose to program in Java then your team may have a performance disadvantage. If you wish to use code from any other source than the specified library page, you must get my approval first.

Rules:

Your agents are subject to the rules described in Section 2.2.2 of the manual. As long as your agent plays by these rules, you are free to design your agent in whatever way you think gives you

the best chance of winning. You are also free to discuss any aspect of the project with other teams, but since you will be competing against them, you may lose advantages by doing so.

Beta demo:

In order to ensure that your team gets off to a good start, I will expect you to give a demo that demonstrates your progress on Thursday, Nov. 18 between 2 and 5pm. The minimum expected level of competence is that your team can score on an immobile opponent. In addition to the demo, your team must hand in a short report outlining its design and progress. This should mention what language(s) you are developing in, and describe your approach to coordination and the intended use of communication (if any). Furthermore, the report should describe how you intend to divide up the work among team members. Note, the demo and report will be graded. Please sign up for a half hour demo slot as soon as possible.

The Lehigh Agents Tournament:

In order to put your agents to the test, we have a tournament in which they face off against each other. For the tournament, the soccer server will be hosted robosoccer, and each team will be running from one of the department's other Solaris machines. The tournament will be single elimination, and the matchups in the first round will be chosen randomly. The tournament is tentatively scheduled for Friday, Dec. 10 from 9-11am (during our scheduled final exam period). All members of your team must be present.

Final Submission:

Your project is due at the start of the tournament. You must provide electronic versions of source code, compiled versions of your files (for Solaris), and a five to ten page report, double-spaced. The report should describe your team's design, and should include a short section on what each person contributed to the project. In addition, each team member will be asked to fill out a team evaluation form in which they rate their own contributions and that of each member on the team. These evaluations will play a factor in your individual grade.

Grading:

Grading will be based on how your team performs in the tournament, the quality of your design, and its originality. The breakdown of grades is as follows:

Beta demo and progress report	20%
RoboCup performance	40%
Final report and design	30%
Individual grade	10%

FAQ

- **Determining the load of a specific machine**
Use top. You'll want a machine with load averages in the 0.01 – 0.02 range.
- **Determining the kind of machine**
Use uname -a (*gives a list of info*), uname -i (*prints kind of machine*), uname -X (*long list of info, includes number of processors*)
- **Determining the memory**
Use top again. It will be listed under Memory, just before the list of processes
- **Europa** (Sun Blade 1000, 4096MB memory, 2 CPUs)