

WP Artificial Intelligence and Software Agents

LaserTag Group Competition: Guidelines

June 20, 2020

This outline describes the process leading up to the LaserTag tournament and the process on tournament day. If you have any questions about these guidelines, please reach out to us in the LaserTag Slack channel.

Tick Submission and Review

- As mentioned on the LaserTag assignment sheet (see “Documentation” directory of the LaserTag repository), each tick method must be submitted by **20:00 CET on 22 June 2020** at the latest.
- **IMPORTANT:** if you explicitly reference agent types (**Green**, **Red**, **Blue**, or **Yellow**) in your tick method, please submit four versions of your code with different type referencing. Since we are unable to determine ahead of time which color your team will be assigned at each stage of the tournament, having four versions of your code will allow us to assign your team a color more easily.
- Each tick method must comply with all the LaserTag rules outlined on the LaserTag assignment sheet and LaserTag documentation. If your code does not follow one of the rules, you will not be able to participate in the tournament. **NOTE:** if there is any uncertainty about the rules, please post a message in the Slack channel so clarification can be provided.
- We will review each tick’s ability to run by performing a few test simulation with it. Our goal is to make sure that no exceptions are produced under the conditions we foresee for the tournament. If your code produces exceptions during testing, we will let you know and work with you, if needed, to resolve the issue until **16:00 CET on 23 June 2020**.

Tournament Setup and Process

- The tournament will begin at **9:00 CET on 24 June 2020** in the Zoom session created by Prof. Clemen (see the course schedule for the meeting details).
- The tournament setup and sequence of games is outlined in the PDF document “tournament_table” (see the “Tournament” directory in the LaserTag repository). The games will take place in the order in which they are numbered.
- A simulation tick is defined as one real-time second. Each game will last 1800 ticks (**Note:** this number is subject to change depending on the time it takes for one simulation to finish).

Map description

- The maps that will be played on during the tournament can be found in the “Maps” directory of the repository.
- As shown in the “tournament_table”, each game will be played by either three or four teams.
- A game with three teams will be played on a triangle-shaped map.
- A game with four teams will be played on a rectangle-shaped map.
- The borders of each map are lined with **Barriers** agents, forcing the agents to remain within the bounds of the arena.
- There will be two types of maps:
 - “open-style”: open terrain with different objects of interest (OOI), similar to the map description found in the LaserTag documentation
 - “labyrinth-style”: less open terrain with more tight spaces and corridors

Topics for Discussion

We encourage an open conversation and discussion among the group. Feel free to bring and share your thoughts, opinions, and feedback on the following subjects (as well as anything else related to artificial intelligence, software agents, and your experience with this course and LaserTag):

- Your agents and their AI:
 - What are the main ideas/priorities that guided your design choices?
 - What are your agents’ behaviors driven by?
 - What decisions are your agents programmed to make, and why?
 - What decisions could they have been programmed to make but were not, and why?
 - Which parts of the LaserTag environment do they interact with in what ways?
 - What were some of your challenges (logical and technical) during the development process? How did you overcome those challenges?
- The LaserTag framework:
 - What did you or didn’t you like about this assignment and working with the LaserTag framework?
 - Is there any functionality (or access to already defined methods) that isn’t currently available and that you would have liked to have in the game?
 - Did the provided “USER METHODS” sufficiently compensate for the limitations put in place by forbidding use of most of the MARS DSL keywords. If not, how so and what should be added to make the design process smoother and the game more enjoyable?
 - Do you have any suggestions for improving the LaserTag framework, documentation, or overall project?
 - Do you have any suggestions for the MARS DSL or MARS in general?