2nd Project

- Implement an agent for controlling Ms Pac Man (**not** the ghosts) in the framework described in: http://www.pacmanvghosts.co.uk/index.html
- Unfortunately, the site is not maintained anymore and the code as well as some pictures are missing. However, I am confident that you can still get the basics and proceed with the assignment (in case of trouble, email me). The code is included as a zip file in this package.
- Your code shall implement a Behavior Tree AND a Finite State Machine (two implementations) that return in every instance the move command argument for Ms PacMan
- Observe the Partial Observability rule that is imposed in this competition
- To start, read the Guide http://www.pacmanvghosts.co.uk/guide_home.html
- You can use any helper function provided in the API http://www.pacmanvghosts.co.uk/guide_api.html
- Sample codes are provided, essentially you shall modify the code in the class MyPacMan (see the image included in the package) \src\main\java\examples\StarterPacMan\MyPacMan.java There is already code (with bugs, I believe) controlling Ms Pac Man in this class, read it to see how it works, then modify it.
- I recommend implementing everything from scratch (translating the FSM /BT into code, since that will be much faster and easier for you. However if you wish, you are free to use any Behavior Tree or FSM library you can find online (example: https://github.com/libgdx/gdx-ai).
- Teams of 2. Submission deadline: Monday after the exams (15/2). Oral exam will follow.
- Submit code as well as a document describing your approach and your BT / FSM (include simple drawing for BT / FSM). Run your code 10 times for each approach and include in your document the average score (S) and average time (T).
- Implementing other approaches (e.g. learning based) would add marks. There are some examples in the code already.