

## 2<sup>nd</sup> Project

- Implement an agent for controlling Ms Pac Man (**not** the ghosts) in the framework described in: <http://www.pacmanvghosts.co.uk/index.html>
- Your code shall be implement a Behavior Tree and/or a Finite State Machine that returns in every instance the move command for Ms PacMan
- Observe the Partail Observability rule that is imposed in tis competition
- To start read the Guide [http://www.pacmanvghosts.co.uk/guide\\_home.html](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](http://www.pacmanvghosts.co.uk/guide_api.html)
- Add your code in the framework as described here [http://www.pacmanvghosts.co.uk/guide\\_pacman.html](http://www.pacmanvghosts.co.uk/guide_pacman.html)
- Sample codes are provided, for example  
`\src\main\java\examples\StarterPacMan\MyPacMan.java`
- You are free to use any Behavior Tree or FSM library you can find online (example: <https://github.com/libgdx/gdx-ai>) . But I recommend implementing everything from scratch, since that will be much faster and easier for you.
- Teams of 2. Submission deadline: week after the exams. Submit code as well as a document describing your approach and your BT / FSM. Oral exam will follow
- Implementing both a FSM and a BT will add marks. Same for implementing other approaches (e.g. learning based)
- Using other similar frameworks such as the one described here <http://gameaibook.org/exercises/> (that does not impose the partial observability rule) is allowed after my permission (ask).