Tutorial 1:

Multi-agent Systems -- course and the research topic

- solving hard problems in an environment with other reasoning entities
- several derived topics (planning -> multi-agent planning; optimization -> decentralized optimization; …)
- several new and unique topics (game theory, auctions, voting, mechanism design)
    - these topics are closely related to economics and social sciences (study behavior of a group of people)
    - in our case -- agents
    - often (not exclusively) we are focused on the computational aspect
        - how to play in the game?
        - how to bet in an auction?
        - how to design an auction if we want to earn the most?
        - how to aggregate other agents' preferences to present to the user?

What is an agent? Connection between agents and robots -- a good way how to think about agents is to think about robots:

- robots are placed in some environment and react on the changes in the environment, need to deliberatively solve some task, need to communicate with other robots in the team, etc. …
- robots are currently available - you can easily purchase functional robots (vacuum cleaners, quadrotors, …) that you can deploy and use in practice

However, agents are more abstract. They, for example, can also be seen as an abstraction methodology -- consider a multi-agent simulation of a real-world large city and different level-of-detail of simulation based on the chosen autonomy/abstraction level.

Think about the robotic soccer -- how can you design a robotic soccer player?

- simple reactive agent -- (hierarchical) if-then rules
    - what happens? (miscoordination, deadlocks, …)
- coordinated agents
    - what happens? (better, but opponent takes the ball)
- reasoning about opponent
    - what happens? (fine, but intractable (the opponent will score while we are thinking what to do))
- decomposition of the team of cooperative agents

3 main topics/tasks that we will cover over the semester:

- coordination of **cooperative** agents
  - a group of agents must accomplish some common goal in an uncertain environment
  - formal model of reasoning -- modal logic, knowledge system, BDI
- solving problems in environment populated with **self-interested** agents
  - game-theory, solving the game
  - social choice, voting, auctions, mechanism design
- solving problems in cooperative setting, where agents wants distributively solve a common task
  - distributed constraint satisfaction programming
  - optimization

Rules for the tutorials

- Throughout the course we will use Java, MasSim, IBM CPLEX
  - Java -- simple, easy to debug, you should know it
  - MasSim -- simple environment that offer basic 'multi-agent robotic-like' environment
    - imperfect sensors
    - imperfect actuators
    - multiple agents
  - CPLEX
    - one of the best MILP/LP solvers
    - sure there are others, but this is not a competition :)
- Programming skills are required -- programming agents in a multi-agent environment involve problems from concurrent programming:
  - accessing shared memory
  - sending messages and their asynchronous parsing
  - outdated information about the environment
- Earning points for homework assignments
  - 9p + 14p + 17p = 40p all together, at least 20p for gaining the assessment
  - you will upload your solution via upload system
    - semi-automatic evaluation
  - deadlines are **strict!**