Project Guide 01

Fundamentals of Artificial Intelligence

MSc in Applied Artificial Intelligence, 2025-26

Version: 2025-10-06

Introduction

- The goal of the project is to develop an intelligent agent, documenting the design and implementation steps

- Each work group needs to be made up of **2 students**

- The project must be available in a Github repository
  - The code and documentation must be merged into a Jupyter Notebook
  - The repository must contain all files required to run the solution
  - The repository must be shared with the lecturer

- A ZIP file of the repository must be submitted on Moodle before the deadline

- The project must be later **presented and defended** by the students

Collaborative Platforms

For hosting the project, you have 2 options: **Google Colab** or **GitHub**

Use **Google Colab**

- One of the members of the group must create a folder in Google Drive using the nomenclature "**FA25_G##**", where G## corresponds to the group

- The folder needs to be shared with the group members and the lecturer.

Use the **GitHub platform**

- One of the members of the group must create a private repository
- Use the nomenclature "**FAI25_G##**", where G## corresponds to the group
- Add the other members of the group as collaborators

- The lecturer should also be added to the repository and the project

The project should be submitted in the **Moodle** platform as a **ZIP file**.

Theme — Class Timetable

CSP Problems can have an huge search space

A problem can have many of solutions of different categories:

- A **possible solution** is any solution, a <u>complete assignment</u>, whether or not it breaks any number of constraints.

- A **feasible solution** is a solution that does not break any (negative) hard constraints.

- An **optimal solution** is a solution with the highest score.

- The **best solution found** is the solution with the highest score found by an implementation in a given amount of time.

A problem can include different types of constraints

- **Hard constraints** — mandatory to get a valid solution.

- **Soft constraints** — preferred but not strictly required. so can be violated under certain circumstances.

Classes Timetable

- Each new academic year, the IPCA administrative team faces difficulties in creating class schedules.

- It is necessary to meet all restrictions related to teachers, courses, classrooms, etc.

- The project aims to provide a tool for generating class schedules for undergraduate courses at the higher education institution.

Source: https://meet.nyu.edu/academics/creating-your-ideal-course-schedule-as-an-nyu-student/

6

Timetable example

	Monday	Tuesday	Wednesday	Thursday	Friday
9h00 - 11h00	**Algorithms and Data Structures** Timeslot_1 Room A	**Software Design and Modelling** Timeslot_5 Room G	**Algorithms and Data Structures** Timeslot_9 Room A	**Machine Learning** Timeslot_13 Room A	Timeslot_17
11h00 - 13h00	**Machine Learning** Timeslot_2 Room A	**Object Oriented Programming** Timeslot_6 Lab 02	Timeslot_10	**Software Design and Modelling** Timeslot_14 Room G	**Object Oriented Programming** Timeslot_18 Lab 02
14h00 - 16h00	**Discrete Mathematics** Timeslot_3 Room C	Timeslot_7	**Discrete Mathematics** Timeslot_11 Room C	Timeslot_15	Timeslot_19
16h00 - 18h00	Timeslot_4	Timeslot_8	Timeslot_12	Timeslot_16	Timeslot_20

Hard Constraints

- Classes last 2 hours.

- All classes have 10 lessons per week.

- Each course may have one or two lessons per week.

- A class cannot have more than 3 lessons per day.

- The timetable for a course may be subject to teacher availability.

Soft Constraints

- The lessons of the same course must be in distinct days.

- Each class should have, if possible, only four days of lessons.

- Each day, lessons should be consecutive.

source: https://www.tech4law.co.za/business/practice-management-d58/be-flexible-man/

Challenges

After implemented previous hard and soft constraints, you are invited to try to add some more.

Hard constraints

- When online classes are scheduled, limited to a maximum of three, they must be scheduled on the same day.
- Somes classes are required to be assigned to a specific classroom.

Soft constraints

- The number of classrooms used by each class should be minimised.

source: https://fdr.at/en/project/barrieren-wo-sie-entstehen-wie-wir-sie-abbauen/
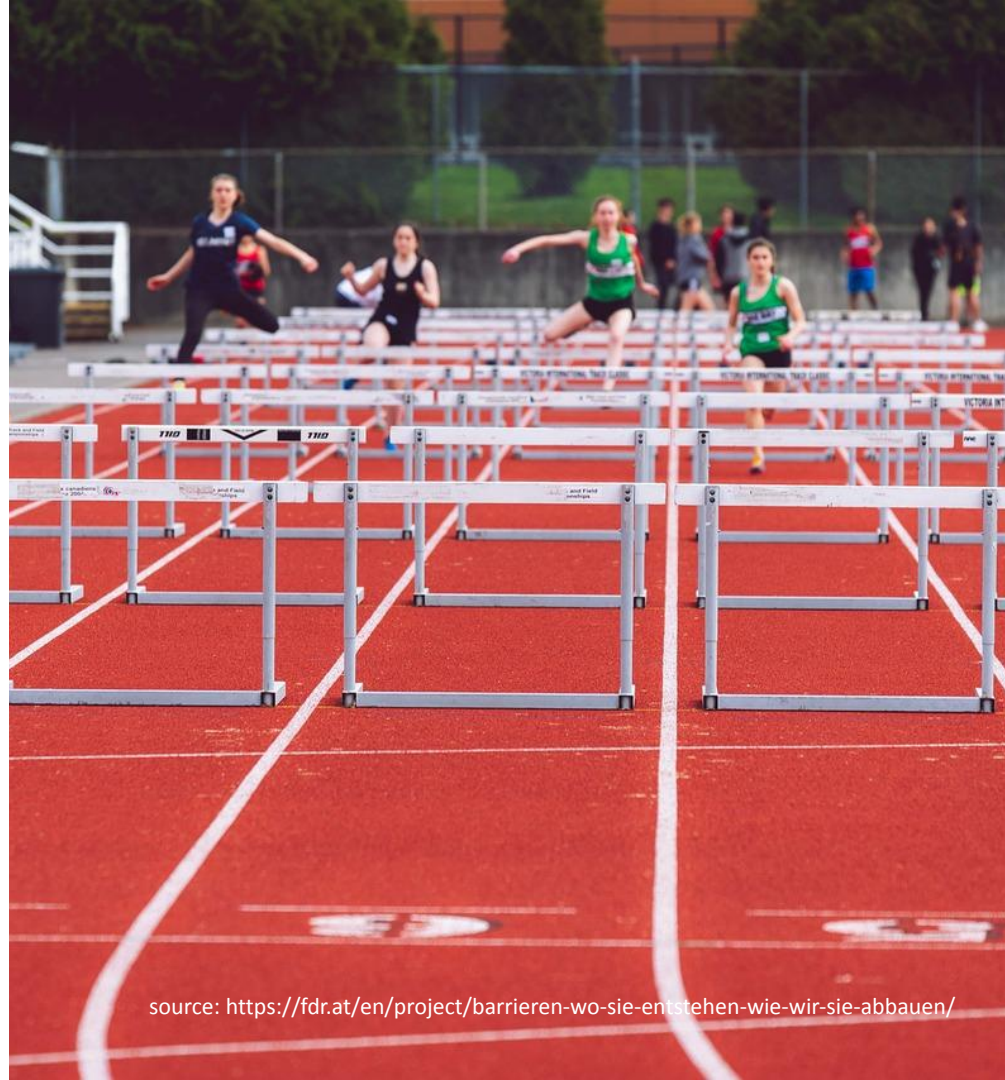
Jupyter Notebook

Tools

- The project must be implemented in Python using a **Jupyter Notebook** and the **Constraint library**. Please check these links:
  - Constraint library site — https://pypi.org/project/python-constraint/
  - Constraint library documentation — http://labix.org/doc/constraint/

- Two datasets will be available in Moodle
  - a small, simpler dataset
  - a larger, more complex one

- The notebook must be well structured
  - Each step in a distinct block
  - Text blocks documenting the reasoning and decisions
  - Comments in the code blocks

Notebook structure

- Introduction
  - Establish here the context and the purpose of project
  - Identify the teammates: student name and number

- Agent design
  - Formulate the problem as a Constraint Satisfaction Problem (CSP)
  - Present the variables, their domain and the constraints definition
  - Highlight the heuristics applied

Notebook structure (2)

- Agent running
  - Find how many solutions exist for the problem
  - Provide the best solution you can find
    - you can use an evaluation function to select the best solution from all available
  - Perform a critical analysis of the results
    - Provide execution times for the solvers used
    - identify some future improvements to the agent

- Conclusion
  - Draw up a conclusion about the outcomes, the development process and the tools used.
  - The structure of the notebook can be adapted according to each project characteristics.

Thank you!