First project proposal: Timetable

Implement a system able to read restrictions for a timetable and check if they can all be satisfied.

The system is able to:

- Read restrictions concerning a user.
- Check if current constraints allow a solution.
- Produce a solution, if one is available.

A graphical UI is optional.

Read restrictions concerning a user

- The restrictions can be inputted either from a graphical UI, the command prompt or from a text/csv file.
- The restrictions have to include a user unique ID, a list of event IDs associated with that user, mandatory or excluded days, mandatory or excluded hours, mandatory or excluded location IDs.
- All of an user's constraints are considered as valid for all of that user's events.
- The system can keep constraints for multiple users. If additional constraints are added for an user with previously known constraints, no editing or compatibility checks are required.
- No login/registration is required. The system will just keep a list of valid user IDs and will only add constraints for them.

Check if current constraints allow a solution

All constraints are considered “hard”.
Additional reading:

- https://drive.google.com/file/d/1kBIpzm5vrSE160dSIf6ANZO4QDfkXt8j/view?usp=sharing
- https://hal.archives-ouvertes.fr/hal-02557389/document
- https://ijcai-15.org/downloads/tutorials/T10-ConstraintLogicProgramming.pdf

If a solution is available, the system confirms this as output.
If no solution can be found, conflicting user ID’s are produced as output. The user is then prompted to remove all constraints from one or more of the conflicting users.

This functionality has to be fully implemented in CLIPS or PROLOG.

Produce a solution, if one is available

Any output associating each of the inputted events for all users with a day, hour and location are accepted:

- PROLOG/CLIPS output, if the above is readable.
- An output text/csv file.
- A graphical UI displaying the solution.

If required, CLIPS/PROLOG wrappers can be used. Suggestions:

CLIPS: clipspy or clipsjni

PROLOG: pylog or projog