German University in Cairo
Department of Computer Science
Dr. Nourhan Ehab

Concepts of Programming Languages, Spring Term 2022
Project 1: Labs Scheduling System
Due: 6th May 2023

1. **Project Description.** In this project you are going to implement a scheduling system in Prolog to assign the weekly labs of a course to the available TAs. The following should be noted:

   -Each lab should be assigned exactly one TA.

   -Each TA has a teaching load (the number of labs they are required to teach per week).

   -Each TA should not be assigned more labs than their teaching load (but can get less load).

   -The number of slots assigned to each TA per day must not exceed a particular specified number.

2. **Required Predicates.** Your implementation must contain the five below predicates. Read the description of all of them before you start your implementation.

   Note that:

   1. You can add any other helper predicates you need.

   2. You can use any predefined predicates **except for** `assert` and `retract`.

   3. It is easier to implement the predicates in the reverse order of how they are listed below.

   a) `week_schedule(WeekSlots,TAs,DayMax,WeekSched)` such that:

   - `WeekSlots` is a list of 6 lists with each list representing a working day from Saturday till Thursday. A list representing a day is composed of 5 numbers representing the 5 slots in the day. The number at position i in a day list represents the number of parallel labs at slot i.

     Example:

     ```
     WeekSlots = [ [0, 0, 0, 0, 0], [2, 1, 2, 3, 0],
     [2, 0, 1, 2, 0] , [0, 1, 1, 0, 0] , [1, 0, 0, 2, 2] ,
     [2, 1, 3, 1, 0] ]
     ```

     The first list represents that Saturday has no scheduled labs. The second list represents that Sunday has 2 labs in the first slot, a lab in the second slot, 2 labs in the third slot, 3 labs in the fourth slot, and no labs in the fifth slot. And so on ...

   - `TAs` is a list of structures of the form `ta(Name,Load)` where `Name` is the name of the TA and `Load` is an integer representing their teaching load.

Example:

```
TAs = [ta(y, 4), ta(h, 7), ta(r, 8), ta(s, 8)]
```

This means that the course has four teaching assistants namely; `y`, `h`, `r`, and `s`. `y` should teach 4 labs a week, `h` should teach 7 slots a week, `r` should teach 8 slots a week, and `s` should teach 8 slots a week.

- `DayMax` is the maximum number of labs a TA can be assigned per day.

  Example:

  ```
  DayMax = 3
  ```

- `WeekSched` is the weekly assignment of TAs to the labs. It is represented as a list of 6 lists. Each list represents a working day from Saturday to Thursday. Position *i* in a day list is a list containing the names of the assigned TAs to slot *i* in the day.

`week_schedule/4` succeeds if `WeekSched` is a possible assignment of the labs to the teaching assistants in `TAs` according to `WeekSlots` so that none is assigned more than their teaching load or assigned more than `DayMax` labs per day.

Example Query:
Assume that `WeekSlots`, `TAs`, and `DayMax` are substituted in the below query with the values given in the examples above. `WeekSched` is the only variable in the below query.

```
?- week_schedule(WeekSlots,TAs,DayMax,WeekSched).
    WeekSched = [ [[], [], [], [], []],
                  [[r, y], [y], [r, h], [r, h, y], []],
                  [[h, y], [] , [h]   , [h, s]   , []],
                  [[], [s], [s], [], []],
                  [[s], [], [], [s, r], [s, r]],
                  [[h, r], [s], [r, h, s], [r], []]
            ]
```

Since there are no labs on Saturday, no TAs are assigned on Saturday. Since there are two labs on Sunday first slot, the first slot is assigned two TAs `r` and `y`. The second slot is assigned only `y` since it has only 1 lab, and so on. The schedule does not assign any of the TAs more than their teaching load, and does not assign anyone more than 3 labs a day.

b) `day_schedule(DaySlots,TAs,RemTAs,Assignment)` such that:

- `DaySlots` is a list of 5 numbers representing the number of parallel labs in the 5 slots of the day.
- `TAs` and `RemTAs` are lists of TA structures.
- `Assignment` is a list of lists of TA names in `TAs` representing the assignment of the day.

`day_schedule/4` succeeds if `Assignment` is a possible day assignment given the available `DaySlots` and list of course `TAs`, while `RemTAs` is the list of updated TA structures after the day assignment.

Example Query:

```
?- day_schedule([2, 1, 2, 3, 0], [ta(y, 4), ta(h, 7), ta(r, 8), ta(s, 8)],
                                                        RemTAs,Assignment).
RemTAs = [ta(y, 1), ta(h, 5), ta(r, 5), ta(s, 8)]
Assignment = [[r, y], [y], [r, h], [r, h, y], []]
```

Since y was assigned 3 lab in the day, y's load is decremented by 3. Similarly, h's load is decremented by 2, and r's load decremented by 3. s's load is not decremented since s was not assigned any slots in this day.

c) max_slots_per_day(DaySched,Max) such that:

- DaySched is a day schedule showing the assignment of the TAs in every slot.
- Max is a number showing the maximum amount of labs a TA can be assigned in a day.

max_slots_per_day/2 succeeds if no TA is assigned more than Max labs in DaySched.

Example Query:

```
?- max_slots_per_day([[y, h], [y], [r, s], [r, s, h], []], 1).
false

?- max_slots_per_day([[y, h], [y], [r, s], [r, s, h], []], 3).
true
```

d) slot_assignment(LabsNum,TAs,RemTAs,Assignment) such that:

- LabsNum is a number representing the amount of parallel labs in this slot.
- TAs is a list of TAs structures.
- RemTAs is the updated list of TAs structures after the assignment to this slot.
- Assignment is a list of the names of TAs in TAs assigned to this slot.

slot_assignment/4 succeeds if Assignment is a possible assignment to a single slot with LabsNum labs and RemTAs is the list of modified TAs after the assignment.

Example Query:

```
?- slot_assignment(3, [ta(y, 4), ta(h, 7), ta(r, 8), ta(s, 8)],
                                                RemTAs,Assignment).
RemTAs = [ta(s, 7), ta(y, 3), ta(h, 6), ta(r, 8)]
Assignment = [s, y, h]
```

Since s, y, and h are assigned to this slot, their remaining teaching loads are decremented in RemTAs.

e) ta_slot_assignment(TAs,RemTAs,Name) such that:

- TAs and RemTAs are lists of TA structures
- Name is a name of a TA in TAs.

ta_slot_assignment/3 succeeds if RemTAs is the list of TA structures resulting from updating the load of TA Name in TAs.

Example Query:

```
?- ta_slot_assignment([ta(y, 4), ta(h, 7), ta(r, 8), ta(s, 8)],RemTAs,y).
RemTAs = [ta(y, 3), ta(h, 7), ta(r, 8), ta(s, 8)]
```

3. Teams. You are allowed to work in teams of four members. You must stick to the teams submitted in the team submission form. IDs for the submitted teams will be posted on the CMS.

4. Deliverables. You should submit a single .pl file named with your team ID. The submission link will be posted on the CMS prior to the submission.