LINGI2365 : Constraint Programming

Assignment 4 :

Modeling

F.Aubry & Y. Deville

March 2014

1 Objective and practical details

The goal of this assignment is for you to explore how to efficiently model more complex Constraint Satisfaction Problems.

You should respect the following constraints

- Deadline : **Friday 04 April 2014 2pm**.
- This assignment is **mandatory**.
- This assignment must be completed **by groups of two** (the same groups as in the previous assignment).

Modalities

- A **hardcopy of your report** must be turned in (box in front of the INGI secretary) before 2pm on the 04 of April.
- Submit your files, **code and report pdf**, on iCampus in a zip (or tar.gz) with the name **tp1_gXY** where **XY** is your group number.
- In case of problems, write to `f.aubry@uclouvain.be`.

2 Problems

2.1 Read this first

In this assignment you will have to spent some more time on the report. You will have to explain your models. This means explaining what your decision variables are. It also means explaining what constraints you use in your model, and explaining what the effect of these constraints is in terms of your problem.
Example: suppose you are asked to model the NQueens problem, and in your model you have a constraint `alldifferent(q);`. It is not sufficient to write that you use alldifferent so all variables in array `q` take different values. You should **explain** that you use alldifferent to force all the queens on a different row.

2.2 Louvain-La-Neuve Golfer Problem (12 pts)

The coordinator of the Louvain-la-Neuve golf club has come to you with the following problem:

In the club, there are 32 golfers each of whom play golf once a week, and always in groups of 4. He would like you to come up with a schedule of play for these golfers, to last as many weeks as possible, such that no golfer plays in the same group as any other golfer on more than one occasion.
Consider the problem as a Constraint Satisfaction Problem with parameter $n = \#\text{weeks}$.

You are asked to:

- Explain which symmetries can arise for this problem. (2)
- Describe two possible models for this problem.
  - explain your models in detail (2)
  - explain how you can modify your models to take symmetries into account (2)
- Implement both models (considering symmetries) in Comet. (1)
- Explain which variable / value ordering heuristics you use with each model. (1.5)
- What is the theoretical maximum number of week in a schedule? (0.5)
- Indicate the maximum number of weeks you could identify in a reasonable time limit using both models. (1)
- Indicate for each model, for each number of weeks the time needed to find a solution, the number of failures and the number of choices. Explain the results, do they correspond to what you would have expected? (2)

2.3 The Time Tabling Problem (8 pts)

A set of lectures need to be scheduled in the Ste Barbe rooms. Each lecture will be followed by a number of students and requires different features in the room (e.g. beamer, blackboard, microphone, ...) as well as a sufficient room size. You need to assign each lecture to a room and a time slot.

You are given the following information in the problem instance:

- Number of rooms, features, lectures and students

- For each room the number of students this room can hold
- For each student the set of lectures he has to attend
- For each room the set of features in this room
- For each lecture the set of necessary features

You need to assign a time slot t (out of a total of **45** possible ones) and a room r to each lecture l. For a solution to be feasible the following constraints need to be respected:

- A student can attend only one lecture at any time slot
- A room can hold only one lecture at any time slot
- A lecture can take place only in a room having the required features and big enough to hold all the students that need to attend

You are asked to:

- Explain which symmetries arise in this problem. (2)
- Design an efficient model for this problem. Note that in order to obtain an efficient model you will have to use a **table constraint** to link rooms, slots and an id for each room/slot pair. Take a look at the Sport Scheduling example in the course slides.
- Explain your model for this problem and explain how it handles symmetries. (2)
- Implement your model in Comet. (1)
- To solve this problem you will need a search procedure that is more efficient than a simple label. You might want to first consider events that have few room and/or time slot options left. Feel free to test different variable or value ordering heuristics. (1)
- Test your model on each of the instances provided on the iCampus site. A description of their format is on the iCampus site. Indicate for each instance the time needed to solve it, the number of failures and the number of choices. (2)