Problem Solving and Search in AI
Tutorial 3 (on May 19th)

Sarah Gaggl
SS 2014

Exercise 3.1:
Model and solve the peer-review procedure with ASP. For scientific conferences, researchers submit their papers which are reviewed by other researchers. The problem of assigning referees for submissions to a conference is typical for the area of computer science.

Part A:
Construct a program check.lp which checks, given an assignment of submissions to members of the program committee (PC), where the following conditions hold:

1. each PC-member is assigned with at most five submissions;
2. no PC-member is assigned more than three papers that he or she rated with "I don't want to review this paper";
3. no PC-member can rate a submission with different bids;
4. no PC-member is assigned a paper that he or she rated with "I cannot review this paper";
5. each submission is assigned to at least one PC-member who rated the paper with "I am willing to review this paper" or higher;
6. If a PC-member does not bid on a certain paper, by default "I don’t want to review this paper" is assumed as the PC-member’s bid on this paper.

The bids on the papers range from 0 to 3 with the following meanings:

0: "I cannot review this paper",
1: "I don't want to review this paper",
2: "I am willing to review this paper",
3: "I really want to review this paper".

The given assignment of submissions to referees is assumed to be stored in some input files containing:
- \textbf{pc}(M): M is a member of the PC;
- \textbf{paper}(P): P is a submitted paper;
- \textbf{bid}(M,P,B): PC-member M's bid on paper P, where B is a constant from \{0, 1, 2, 3\};
- \textbf{assigned}(P,M): the submission P is assigned to PC-member M.

The program \texttt{check.lp} should satisfy the following condition:

- \texttt{check.lp}, together with the input data, possesses an answer set precisely when Conditions 1.-6. are met.

a) You can use the whole language constructs offered by \texttt{gringo/clasp}.

b) Formulate the same program without the use of aggregate functions.

c) Test your program with three different input instances.

\textbf{Part B:}

Now construct a program \texttt{guess.lp} which assigns, given a collection of submissions and a given PC, the submissions to the members of the PC in such a way that the following condition is satisfied:

\textit{(\ast)} each submission is assigned to exactly three members of the PC.

Use the above defined predicates \texttt{pc(M)}, \texttt{paper(P)} and \texttt{assigned(P,M)}. 