Exercise 6.1:
Given the program $P_1$, determine the stable models of $P_1$ by applying the Gelfond-Lifschitz-Reduct.

$$P_1 = \{ a \leftarrow b, \mathit{not} \, c, d; \\ c \leftarrow \mathit{not} \, b, a; \\ b \leftarrow \mathit{not} \, c, \mathit{not} \, d; \\ a \leftarrow \}$$

$$P_2 = \{ a \leftarrow b, \mathit{not} \, c; \\ b \leftarrow c, \mathit{not} \, a; \\ c \leftarrow a, \mathit{not} \, b; \\ b \leftarrow \}$$

$$P_3 = \{ a \leftarrow \mathit{not} \, b, c; \\ c \leftarrow \mathit{not} \, a, b \}$$

Exercise 6.2:
Recall the 3-Colorability problem (c.f. Exercise 1.1 of FCP). Define an ASP encoding for an arbitrary (undirected) graph $G = (V, E)$, such that each answer-set of your encoding corresponds to a 3-coloring of the graph $G$.

Exercise 6.3:
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:

- \( \text{pc}(M) \): \( M \) is a member of the PC;
- \( \text{paper}(P) \): \( P \) is a submitted paper;
- \( \text{bid}(M,P,B) \): PC-member \( M \)’s bid on paper \( P \), where \( B \) is a constant from \( \{0,1,2,3\} \);
- \( \text{assigned}(P,M) \): the submission \( P \) is assigned to PC-member \( M \).

The program \( \text{check.lp} \) should satisfy the following condition:

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

**Important:** Do not use any aggregate functions for constructing the program \( \text{check.lp} \)!

**Part B:**

Now construct a program \( \text{guess.lp} \) 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:

\( (*) \) each submission is assigned to exactly three members to the PC.

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