Please checkmark exercises that you solved before ***11.01.2018****. The details of the checkmarking process will be available on the course website from* ***11.12.2017*** *onwards. Be sure to tick only those exercises which you can solve and explain on the blackboard. Do not leave the exercise work for the very last moment. Start preparing solutions as early as possible!*

Problem 1. Let P_1 be the following problem:

$$\begin{array}{rl} brother\_of(hans, bob). & brother\_of(hans, luis). \\ sister\_of(mary, bob). & sister\_of(mary, luis). \\ brother\_of(nick, robin). & sibling\_of(ann, hans). \end{array}$$

$$\begin{aligned} sibling\_of(X, Y) &\leftarrow brother\_of(X, Y). \\ sibling\_of(X, Y) &\leftarrow sister\_of(X, Y). \\ has\_sister(Y) &\leftarrow sister\_of(X, Y). \\ has\_only\_brothers(Y) &\leftarrow brother\_of(X, Y), \text{not } has\_sister(Y). \\ relative\_of(X, Y) &\leftarrow sibling\_of(X, Y). \\ relative\_of(X, Z) &\leftarrow sibling\_of(X, Y), relative\_of(Y, Z). \end{aligned}$$

Define the *Herbrand Universe* $HU(P_1)$, the *Herbrand Base* $HB(P_1)$ and three *Herbrand Models* I for P_1. What is the least model of P_1? For the definition of $HB(P_1)$ it is not necessary to explicitly enumerate all atoms but the range of all atoms has to be clear.

Problem 2. Let P_2 be the following normal logic program., where c and d are constants and X, Y are variables.

$$\begin{aligned} selected(X) &\leftarrow available(X), \text{not } not\_selected(X). \\ not\_selected(X) &\leftarrow available(X), \text{not } selected(X). \\ choice\_made &\leftarrow selected(X). \\ &\leftarrow \text{not } choice\_made. \\ available(c). & \\ available(d). & \end{aligned}$$

- Compute the grounding $grnd(P_2)$ of the program P_2.
- Formally check by computing the *Gelfond-Lifschitz reduct* whether the interpretation $I = \{available(c),\ available(d),\ selected(c),\ not\_selected(d),\ choice\_made\}$ is a stable model of P_2.

Problem 3.

1. Is the intersection of two Herbrand models of a normal logic program P is again a Herbrand model? If yes, prove it, otherwise provide a counterexample.

2. Is the union of two Herbrand models of a positive logic program P again a Herbrand model? Again, if yes, prove it, otherwise provide a counterexample.

Problem 4. Let a, b be constant symbols and X, Y be variables. Consider

$$\begin{aligned} q(a). \quad & r(b). \\ s(X, Y, Z) \leftarrow\ & q(X),\ r(Y),\ f(Z). \\ f(X) \leftarrow\ & q(X),\ not\ r(X). \end{aligned}$$

(i) Compute the grounding $grnd(P)$ of P.

(ii) Decide using the *Gelfond-Lifschitz reduct* whether the interpretation $I = \{q(a), r(b), f(a), s(a, b, a)\}$ is a stable model of P.

Problem 5. Define a *normal logic program* P consisting of the ground atoms q, r, f and s that exhibits the following properties.

- P has exactly 3 answer sets,
- $\{q, s\}$ is an answer set of P, and
- P contains no more than 1 fact.

Problem 6. Consider a small fragment of the program for solving the *Project-Assignment Problem* (i.e., assigning managers to various industrial projects)

- Managers are represented by facts of the form
  $manager(aaron)$. $manager(bill)$. $manager(charlotte)$. etc.
- Projects are represented by facts of the form
  $project(p1)$. $project(p2)$. $project(p3)$. etc.
- Assignments of projects to managers are represented by facts of the form
  $assigned(P, M)$, meaning that the project P is assigned to the manager M.
- The managers can express their competence for leading a specific project. The preferences range from 0 ("I am not competent in the subject area of the project") to 3 ("I am really competent and want to lead the project"). The preferences are defined by facts of the form $pref(M, P, B)$ with the meaning that the manager M assessed his competence for the project P with B, where B is a constant from $\{0, 1, 2, 3\}$ with the following meanings:

0: "I am not competent in the subject area of the project",
1: "I can lead the project, but not particularly willing to",
2: "I am willing to lead this project",
3: "I really want to lead this project".

Your tasks are the following.

- Use the syntax of `DLV` and aggregate atoms to define a predicate $count(M, C)$ that counts for a specific manager M all projects that are assigned to M and stores this value in C.
- Use the above defined predicate $count(M, C)$ to specify the following constraints:
  (i) *Projects assigned only to managers who rated their competence in the respected subject area with 0 should be completely excluded.*
  (ii) *For any manager who got at least one project assigned to him/her, the sum of his/her preferences for the assigned projects should be greater or equal than twice the number of projects assigned to that manager.*

  Again use the syntax of `DLV` and aggregate atoms for the definition of this constraint.

Problem 7. Imagine a transport network represented using facts $link(c, d)$, where c, d denote bus stops and $link(c, d)$ states that there is a direct bus connection from c to d. Define a normal logic program that uses a predicate $not\_accessible(c, d)$ to calculate all vertices d that are not accessible from c. (A node d is not accessible from a node c if there is no direct connection from c to d, and if there is no bus route from c to d in the network. A bus route in a network is a sequence of bus stops such that from each of these bus stops there is a direct connection to the next bus stop in the sequence.)

Problem 8. For a program P, we denote by $AS(P)$ the set of all answer sets of P. Let P, Q be programs. We say that P, Q are

- equivalent, if $AS(P) = AS(Q)$ and
- strongly equivalent if $AS(P \cup R) = AS(Q \cup R)$ for every program R.

Prove or refute that if

1. whenever (ii) holds then also (i) and
2. the converse holds, i.e., whether (i) implies (ii).

Problem 9. Assume that an electrical station has two entrance points represented by the constants $nothern\_entrance$ and $southern\_entrance$. An entrance is accessible if it is not known to be closed. If an entrance is closed then it is definitely not accessible. Use the predicates $accessible(X)$ (entrance X is accessible) and $closed(X)$ (entrance X is closed) and define a disjunctive logic program that has exactly the following answer sets:

$A_1 = \{closed(nothern\_entrance), accessible(southern\_entrance), \neg accessible(nothern\_entrance)\}$
$A_2 = \{closed(southern\_entrance), accessible(nothern\_entrance), \neg accessible(southern\_entrance)\}$

Problem 10. Bob, Alice and Jerry went hiking and they cannot be reached by phone. You are their friend and want to figure out where they could be. You have the following information:

- Every person is either at the lake or in the forest but not both.
- Jerry cannot be at the lake without Alice.
- Alice is for sure not with Bob at the lake.
- At least one person is in the forest.

Define a logic program P and compute its answer sets. Use $forest(X)$ and $lake(X)$ as predicate symbols and *bob*, *alice* and *jerry* as constant symbols.