MASTER IN ARTIFICIAL INTELLIGENCE (UDC - USC - UVigo)
REASONING AND PLANNING exam. January 8th, 2024

Surname: _____

First Name: _____

INSTRUCTIONS This exam covers units 1-6 and is weighted with a maximum of **42 points (pt) from a total of 100 pt** in the whole course (Unit 7 is not covered in the exam and weights 8 pt). For the test, use the original statement sheet and avoid corrections or unclear marking (ask for a new blank sheet if needed). **Completion time = 2 hours**.

— EXAM —

Exercise 1 (20pt). Each question has at least one correct answer and its total score depends on whether you check: some incorrect answer = **-3pt**; all the correct answers = **5pt**; only correct answers, but not all = **3pt**; leaving blank = **0pt**. A total negative score in Exercise 1 counts as 0pt in the rest of the exam.

1.1) Mark those formulas below that are stronger than $p \rightarrow \neg q$ in classical propositional logic:

- ☐ $p \wedge \neg q$
- ☐ p
- ☐ $\neg q$
- ☐ $p \wedge \neg p$

1.2) The logic program P with rules `p :- not q, r.` `r :- p.` is stratified. Mark the rules below that, if they were (individually) added to P, they would make the result a non-stratified program.

- ☐ `p :- r.`
- ☐ `q :- not p.`
- ☐ `q :- r.`
- ☐ `r :- q.`

1.3) Given the following logic program `p :- q.` `p :- not p, not q.`

- ☐ the reduct with respect to $\{q\}$ is the program `p :- q.`
- ☐ the reduct with respect to $\{q\}$ is the program `p :- q.` `p :- not p.`
- ☐ the reduct with respect to $\emptyset$ is the program `p :- q.` `p.`
- ☐ the reduct with respect to $\{p\}$ is the program `p :- q.` `p.`
- ☐ the reduct with respect to $\{p, q\}$ is the program ☐

1.4) The rule `p :- not p, not q.` used above is actually equivalent to the formula $\neg\neg p \vee \neg\neg q$ in the logic of Here-and-There (HT), but the latter is not equivalent to $p \vee q$ in that logic. Mark those HT interpretations that are HT models of $\neg\neg p \vee \neg\neg q$ but not of $p \vee q$.

- ☐ $H = \emptyset$, $T = \{p\}$
- ☐ $H = \{q\}$, $T = \{q\}$
- ☐ $H = \emptyset$, $T = \{p, q\}$
- ☐ $H = \{p, q\}$, $T = \{p\}$

Exercise 2 (10pt). Write an ASP program that generates all ways to place 5 rooks in a chessboard so that they do not attack each other. Use predicate `rook(X,Y)` meaning there is a rook at row `X` and column `Y`. (NOTE: in chess, rooks attack other pieces in the same row or in the same column).

```
#const n=8.
cell(1..n,1..n).









#show rook/2.
```

Exercise 3 (8pt). The following `telingo` program tries to move a robot in a grid from an initial position at `(0,0)` to a goal position at `(3,4)`. Complete the program to fulfil the two missing requirements: (1) the robot must reach the goal position at the end; (2) the robot cannot step into a wall.

```
#program initial.
grid(0..3,0..4).
wall(0,2). wall(2,2).  wall(3,2).   robot(0,0).  goal(3,4).

#program dynamic.
1 { robot(X+1,Y); robot(X-1,Y); robot(X,Y+1); robot(X,Y-1) } 1 :- 'robot(X,Y).
:- robot(X,Y), not _grid(X,Y).  % Do not step out of the grid
```

Exercise 4 (4pt). Write a formula in Description Logic (DL) that describes the set of individuals with children (use relation *has_child*) being all of them students (use concept name *Student*).