Exam for IN4010TU Artificial Intelligence Techniques

3 November 2010

This exam will test your knowledge and understanding of the material discussed in the first period of the course Artificial Intelligence Techniques. Using the book, lecture notes, or slides during the examination is *not* allowed. You will have 3 hours (from 14 till 17) to complete the exam. It has 4 questions, for a total of 59 points. Please don't include irrelevant information: you will be marked down for this. Before you hand in your answers, please check that you have put your name and student number on top of every sheet you hand in.

Questions

Question 1 *9 points*

(a) (3 points) Explain what a *rational agent* is.

(b) (3 points) Explain what the cognitive modeling approach towards Artificial Intelligence is. Provide two reasons why you think this approach is useful or not.

(c) (3 points) Name one difference between the Turing Test approach and the Rational Agent approach towards Artificial Intelligence.

Question 2 *22 points*

This question is about agent programs, Prolog, and GOAL.

```
main:ferryAgent
{
   knowledge{
     transportedCars :- ... .
     otherSide(X,Side) :- ... .
   }
   beliefs{
     side(north). side(south).
     car(ferrari). at(ferrari,north).
     car(toyota). at(toyota,north).
     car(opel). at(opel,north).
     at(ferry,north).
   }
   goals{ transportedCars. }
   program{
     if goal(transportedCars), bel(at(X,north)) then board(X).
     if goal(transportedCars) then sail.
     if goal(transportedCars), bel(at(ferry,south)) then disembark.
   }
   actionspec{
     board(X){
       pre{ at(X,Y), side(Y), car(X), at(ferry,Y) }
       post{ not(at(X,Y)), on(X,ferry) } }
     sail{
       pre{ at(ferry,X), side(X), otherSide(ferry,Y) }
       post{ ... } }
     disembark{
       pre{ ... }
       post{ at(X,Y) , not(on(X,ferry)) } }
     }
   }
}
```

Figure 1: GOAL Agent Program

Consider the agent program listed in Figure 1. The agent's task is to transport all cars from the north side of a river to the south side of that river by controlling a ferry. Cars can be boarded onto and disembarked from the ferry when the car is located at the same side of the river as the ferry. In your answers to the questions, do not introduce new predicates that have not been mentioned either in the program or this question.

(a) (3 points) Provide a definition for the predicate `transportedCars`. `transportedCars` should follow if and only if all cars have been transported to the `south` side. (Hint: Use the `forall(Cond1, Cond2)` operator. As an example to illustrate the use of this operator, consider that the query: `forall(on(X,ferry), car(X))` succeeds if all things that are on the `ferry` are cars.)

(b) (3 points) Similarly, provide a definition for the predicate `otherSide(X,Side)` which holds if and only if `X` is a car or ferry and `Side` is the opposite side of where the car or ferry is located (i.e. south if `X` is at the north side, and vice versa).

(c) (4 points) List the actions that can be executed by the GOAL agent given the agent's belief and goal base listed in Figure 1.

(d) (3 points) Complete the action specification for the action `sail`.

(e) (3 points) Complete the action specification for the action `disembark`.

(f) (6 points) One of the desirable properties of the agent is that it gets each car to the other (south) side. For the agent listed, since we know there is a ferrari car at the north side, we can express this by:

$$\Diamond \texttt{bel(at(ferrari,south))}$$

Explain why the agent satisfies this property, or does not satisfy it. You do not have to provide a listing of all traces but may provide an informal argument and describe a trace informally.

Question 3 *20 points*

Consider the following planning problem that involves a household robot that prepares dinner.

Init(havePotatoes)

Goal(tableLaid ∧ haveBakedPotatoes ∧ haveBakedFish)

Action(bakePotatoes,
PRECOND: havePotatoes ∧ ovenHot
EFFECT: ¬havePotatoes ∧ haveBakedPotatoes)

Action(buyFish,
PRECOND:
EFFECT: haveFish)

Action(cookFish,
PRECOND: haveFish ∧ handsClean
EFFECT: ¬haveFish ∧ haveBakedFish)

Action(heatOven,
PRECOND:
EFFECT: ovenHot)

Action(layTable,
PRECOND: handsClean
EFFECT: tableLaid)

Action(washHands,
PRECOND:
EFFECT: handsClean)

(a) (15 points) Construct a solution for this planning problem using partial order planning.

(b) (5 points) Planners based on planning graphs use so-called *mutex links* to identify mutual exclusion between ~~among others~~ actions. A mutual exclusion relation between two *actions* holds if there are (i) *inconsistent effects*, (ii) *interference*, and/or (iii) *competing needs* between these actions. Explain each of these three conditions (i-iii), i.e. provide a description of what an inconsistent effect is, what interference is, and what competing needs are.

Question 4 *8 points*

(a) (3 points) Explain the difference between *active* and *passive* sensors. Provide an example of a passive and of an active sensor.

(b) (5 points) Explain why uncertainty, e.g. the fact that a robot does not know exactly where it is, which is a characteristic problem associated with robotics, complicates the robot motion planning problem. Provide one simple solution.

End of exam