UNIVERSITY OF DUBLIN
TRINITY COLLEGE

Faculty of Engineering, Mathematics and Science

School of Computer Science & Statistics

Integrated Computer Science
B.A. (Mod.) CSLL
B.A. (Mod.) Business & Computing

Trinity Term 2013

Artificial Intelligence I

Saturday 18/05/2013 **GOLDHALL (220)** **9:30-11:30**

Dr Tim Fernando

Instructions to Candidates:

Attempt **two** questions (out of the three given).

All questions carry equal marks. 50 marks per question.

You may not start this examination until you are instructed to do so by the Invigilator.

Materials permitted for this examination:

Non-programmable calculators are permitted for this examination — please indicate the make and model of your calculator on each answer book used.

1. Recall that a *definite clause* is an atom (or fact) or a rule of the form

```
h :-  b1,b2,...,bm
```

where `h` and all `bi`'s are atoms.

(a) What does it mean for a definite clause to be a *logical consequence of* a set KB of definite clauses?

[5 marks]

(b) A *propositional clause* is a definite clause in which all predicates in it have arity 0. (That is, there are no terms.) Let us agree to encode propositional clauses as lists, with an atom `f` encoded as `[f]` and a rule `h :- b1,..,bm` as `[h,b1,...,bm]`. A finite list of propositional clauses can then be encoded as a list of lists — e.g.

```
h :- c.
h :- f,g.
f :- g.
g.
```

as `[[h,c], [[h,f,g], [f,g], [g]]`. Now, consider the binary Prolog predicate `lc(Atom,KB)` defined below.

```
lc(Atom,KB) :- member([Atom|B],KB), lcAll(B,KB).

lcAll([],_]).
lcAll([H|T],KB) :- lc(H,KB), lcAll(T,KB).
```

Let *kb* be a list of propositional clauses (encoded as lists), and let *a* be a propositional atom.

(i) State if the following is true or false: `lc(`*a*`,`*kb*`)` is true exactly if *a* is a logical consequence of *kb*. Justify your answer.

[5 marks]

(ii) State if the following is true or false: `lc(`*a*`,`*kb*`)` is true exactly if after consulting the clauses encoded by *kb*, the Prolog interpreter says yes (or `True`) to the query *a*. Justify your answer.

[5 marks]

(c) Define a 3-ary predicate `arc(Node1,Node2,KB)` and a unary predicate `goal(Node)` so that the predicate `lc(Atom,KB)` above can be defined equivalently by

```
lc(Atom,KB) :- search([Atom],KB).
search(Node,_) :- goal(Node).
search(Node,KB) :- arc(Node,Next,KB), search(Next,KB).
```

[10 marks]

(d) Assuming each arc in the graph defined in (c) has cost 1, describe a heuristic estimate of the distance of a node to a goal node that is an underestimate.

[5 marks]

(e) Use the cost and heuristic functions in part (d) to define an A-star search on the graph in part (c). Outline the revisions to `search` in part (c) involved in implementing A-star.

[15 marks]

(f) Is your A-star search in part (e) admissible? Justify your answer, explaining what admissibility means.

[5 marks]

2. This question concerns a version of the so-called Zebra Puzzle simplified to (c1)-(c9) below.

   (c1) There is a street with three neighbouring houses: a leftmost one, a middle one, and a rightmost one.

   (c2) Each house has a different colour, namely red, blue, and green.

   (c3) People of different nationalities live in the different houses.

   (c4) Each house has a different pet.

   (c5) The Englishman lives in the red house.

   (c6) The jaguar is the pet of the Spanish family.

   (c7) The Japanese lives to the right of the snail keeper.

   (c8) The snail keeper lives to the left of the blue house.

   (c9) The snail lives in the middle house.

   The question is who keeps the zebra? Before defining a predicate zebra/1 that tells us the nationality of the zebra keeper, let us recall some basic definitions about *Constraint Satisfaction Problems* (CSPs). A CSP is given by a set X of variables, a domain set D_x for each variable x in X, and a set C of constraints on instantiations of the variables in X.

   (a) For the puzzle above, let us work with 9 variables:

   - `Col1,Col2,Col3` for the colors of the leftmost, middle and rightmost houses
   - `Nat1,Nat2,Nat3` for the nationalities of the leftmost, middle and rightmost houses
   - `Pet1,Pet2,Pe3` for the pets of the leftmost, middle and rightmost houses

   Give the domain sets D_x for each of the variables above. How does the *generate-and-test* approach to CSP work in this particular puzzle? Up to how many possibilities might be generated and tested? Explain.

   [10 marks]

   (b) A popular alternative to generate-and-test is to instantiate variables incrementally, starting with a node that instantiates no variable, but moving along arcs from `Node1` to `Node2` only if `Node2` instantiates some uninstantiated variable in `Node1`, while keeping all variable instantiations in `Node1`.

   For the specific case of the puzzle above,

(i) give a bound on the depth for searching that graph

[5 marks]

(ii) give a bound on the number of nodes there are in the graph

[5 marks]

(iii) explain how your answers in (i) and (ii) compare to generate-and-test (part (a)).

[5 marks]

(c) Under the approach outlined in part (b), which of constraints (c4) to (c9) is the most useful to consider when expanding the start node (instantiating *no* variable)? What node results from this constraint? Explain.

[10 marks]

(d) Continue the search in part (c), describing a sequence of constraints to consider that leads a solution to the question: who owns the zebra? How do we choose the constraint to consider and the arcs to explore? When is backtracking necessary?

[10 marks]

(e) How does the instantiation of variables above correspond to building an interpretation (or model)? Be sure to state what an interpretation is in this case.

[5 marks]

3. Consider the knowledge base

```
false :- a,b.
false :- c.
a :- p, r.
a :- q.
b.
```

(a) What is the *Complete Knowledge Assumption* (CKA)? What can we conclude if we apply CKA to the knowledge base above?

[10 marks]

(b) Which of these conclusions are logical consequences of the knowledge base? Which are not?

[10 marks]

(c) What does it mean for an inference system to be *non-monotonic*? State if the following is true or false: CKA leads to non-monotonicity.

[5 marks]

(d) What is an *integrity constraint*, and what is a *Horn clause*? List all integrity constraints and Horn clauses in the knowledge base above.

[10 marks]

(e) Suppose `p,q,r` were assumable in the knowledge base above. What are the conflicts? What are the minimal conflicts?

[15 marks]