FACOLTÀ DI INGEGNERIA DELL'INFORMAZIONE

Artificial Intelligence 2009–10

Test 1 – 29th January 2010

RULES: **You have 2 hours to complete the test**
You can use no texts, written notes, computers, calculators, mobile phones
Please write your name and student ID on all sheets
Please answer Question 3 on a separate sheet to speed up the marking process

1. State Space (10 points)

1.1 Specify the general Tree Search Algorithm (in pseudo-code or in a programming language of your choice).

Then explain how the algorithm is adapted to generate the following search strategies: Depth First, Breadth First, Uniform Cost, A*.

1.2 Explain what it means for a search strategy to be optimal.

Then, again with respect to Tree Search, specify the conditions under which the A* strategy is optimal.

2. Planning (10 points)

Consider the planning problem stated below:

A robot wants to put a book on the table. Initially, the book is on the chair, far from the table, and the robot is at the window, far from both the book and the table. The robot can: walk from any place to any other place; pick up things from any place (if its hand is free); put down things at any place.

Now:

- define a set of predicates (and possibly constants), briefly describing their intuitive meanings;
- define a sufficient set of action schemes;
- represent the initial state and the goal;
- write a plan that achieves the goal;
- describe the states obtained after the execution of each action of the plan.

(You do not have to explain how an automatic planner would build the plan).

3. Logic (12 points)

3.1 Exceptions are basically contradictions, so it is interesting to see how we can model them in logic formulae, which are supposed to be clear and consistent. Formalize the following sentences that include an exception in first order logic formulae:

a) When I am angry, nothing can make me happy except for a cake.

b) Everybody went to the party, except for me.

c) Nobody is allowed on this floor, except for managers and senior researchers.

Did you find a general rule to model the expression "except for"?

3.2 Formalize the following proof in propositional logic and point out the inference rules that have been used step by step:

If I catch the train, then either I'm on time or the train is late. If I wake up late, I must run to be on time. I'm out of shape, which means that if I run I end up heavy breathing. Today the train was on time, while I woke up late, but I managed to catch it. That's why I'm heavy breathing.

Solution of 2

Constants: Book, Table, Chair, Window

Predicates:

On(x,y)	object x is on object y
At(x)	the robot is at x
Has(x)	the robot has object x in its hand
FreeHand( )	the robot's hand is free

Action schemes:

take(x,y) = // the robot takes x from y
- P: On(x,y), At(y), FreeHand( )
- E–: On(x,y), FreeHand( )
- E+: Has(x)

put(x,y) = // the robot puts x on y
- P: Has(x), At(y)
- E–: Has(x)
- E+: On(x,y), FreeHand( )

walk(x,y) = // the robot walks from x to y
- P: At(x)
- E–: At(x)
- E+: At(y)

Initial state: At(Window), On(Book,Chair), FreeHand( )

Goal: On(Book,Table)

Plan with resulting states (the goal is underlined):

start	At(Window), On(Book,Chair), FreeHand( )
walk(Window,Chair)	At(Chair), On(Book,Chair), FreeHand( )
take(Book,Chair)	At(Chair), Has(Book)
walk(Chair,Table)	At(Table), Has(Book)
put(Book,Table)	At(Table), <u>On(Book,Table)</u>, FreeHand( )
end	