FACOLTÀ DI INGEGNERIA DELL'INFORMAZIONE

Artificial Intelligence 2008–09

Test 2 – 19th February 2009

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 (12 points)

1.1 Explain the difference between **tree search** and **graph search**, clarifying: (i) why and when we need graph search; (ii) what improvements are required by the tree search algorithm to implement graph search (you are not asked to describe the whole algorithm, only the relevant modifications).

Now consider the 8-puzzle: does it require graph search? If so, why?

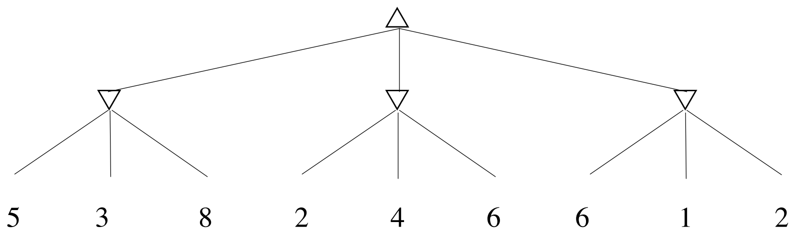
1.3 Suggest a state space representation for the problem stated below: define a data structure for representing the states and specify the allowable actions (you are not required to find a solution).

> *An old Russian puzzle*. While travelling with a wolf, a goat and a (big) cabbage, a farmer has to cross a river from side A to side B. Initially, a small boat is available at side A. The boat can carry at most one object plus the farmer, who obviously is the only one able to row. The farmer has to find a way of bringing the wolf, the goat and the cabbage from A to B, with the following constraints: at no stage of the solution should the wolf be left with the goat in absence of the farmer (because the wolf would eat the goat); analogously, at no stage of the solution should the goat be left with the cabbage in absence of the farmer (because the goat would eat the cabbage).

2. Adversarial search (8 points)

2.1 With respect to the technique known as *alpha-beta pruning*, explain: (i) to what types of games the technique can be applied; (ii) the meaning of the parameters *alpha* and *beta*, and their mathematical relationship; (iii) the meaning of the term *pruning* and the advantage expected by using such a technique.

2.2 Show how alpha-beta pruning would process the following game tree:



3. Logic (12 points)

3.1 Formalize the following sentences into first order logic formulae:

a) only a barber can shave people;
b) there is only one barber;
c) one does not shave oneself;
d) all barbers have a beard, unless they are a woman.

3.2 Transform the first order logic formulae into clauses.

3.3 Prove that the sentence d is a logical consequence of sentences a - c (there may be the need to make some hidden assumptions explicit).

LOGIC SOLUTION:

3.1 Formalize the following sentences into first order logic formulae:

a) only a barber can shave people;

$\forall x \forall y(Shave(x,y) \Rightarrow Barber(x))$

b) there is only one barber;

$\exists x(Barber(x) \wedge \forall y(Barber(y) \Rightarrow y=x))$

c) one does not shave oneself;

$\forall x \neg Shave(x,x)$

d) all barbers have a beard, unless they are a woman.

$\forall x(Barber(x) \Rightarrow (Beard(x) \vee Woman(x)))$

3.2 Transform the first order logic formulae into clauses.

a) $\forall x \forall y(Shave(x,y) \Rightarrow Barber(x))$ becomes

$\neg Shave(x,y) \vee Barber(x)$ [A]

b) $\exists x(Barber(x) \wedge \forall y(Barber(y) \Rightarrow y=x))$ becomes

$Barber(K)$ [B1] and

$\neg Barber(y) \vee y=K$ [B2]

c) $\forall x \neg Shave(x,x)$ becomes

$\neg Shave(x,x)$ [C]

d) $\forall x(Barber(x) \Rightarrow (Beard(x) \vee Woman(x)))$ becomes

$\neg Barber(x) \vee Beard(x) \vee Woman(x)$ [D]

3.3 Prove that the sentence d is a logical consequence of sentences a - c (there may be the need to make some hidden assumptions explicit).

We can prove that formula d is a logical consequence of the rest by refutation with the resolution technique. First, we need to negate the thesis:

d') $\neg \forall x(Barber(x) \Rightarrow (Beard(x) \vee Woman(x)))$ which becomes

$Barber(H)$ [D1] and $\neg Beard(H)$ [D2] and $\neg Woman(H)$ [D3]

Implicit assumptions are:

e) if a man (i.e. an individual who is not a woman) does not get shaved, he has a beard

$\forall x(\neg Woman(x) \wedge \neg \exists y\, Shave(y,x) \Rightarrow Beard(x))$ which becomes

$Woman(x) \vee Shave(f(x),x) \vee Beard(x)$ [E]

The assumption that women do not have beards is actually not necessary for our proof.

B2 + D1 leads to H = K [R0], so that we can rewrite everything about H in terms of K, as they designate the same element in the domain. We get:

$Barber(K)$, which was already in our knowledge base, $\neg Beard(K)$ [D2'], and $\neg Woman(K)$ [D3'].

E + D3' yields $Shave(f(K),K) \vee Beard(K)$ [R1]

R1 + D2' leads to $Shave(f(K),K)$ [R2]

R2 + A gives $Barber(f(K))$ [R3]

B2 + R3 yields $f(K) = K$ [R4] so that R2 can be rewritten into $Shave(K,K)$ [R2']

R2' and C lead to the empty clause, thus proving that d' is incompatible with the other formulae, that is, d is their logical consequence.