UNIVERSITY OF LONDON

IMPERIAL COLLEGE OF SCIENCE, TECHNOLOGY AND MEDICINE

EXAMINATIONS 1999

MSc Degree in Advanced Computing
for Internal Students of the Imperial College of Science, Technology and Medicine

This paper is also taken for the relevant examinations for the Diploma of Membership of Imperial College

PAPER A 4.94

ARTIFICIAL INTELLIGENCE

Thursday, May 13th 1999, 10.00 – 12.00

Answer THREE questions

For admin. only:
paper contains 4 questions

1 a i) What is meant by saying that a search algorithm for finding a minimum cost solution path in a problem solving search graph is *admissible*?

ii) What value is associated with each node N of the exploration of a search graph by the A* graph search algorithm? What node does A* select for expansion at each step in the construction of its search tree.

iii) What are the two conditions on the use of the A* algorithm that guarantee admissibility? Using one of these conditions, prove that any solution path found by A* will be a minimum cost path.

b Briefly compare the heuristic search approach to problem solving with the planning approach using STRIPS style action descriptions.

c A kitchen robot needs to plan its cleaning activity. Give the STRIP actions descriptions for the following cleaning actions and their effects:

i) Cleaning the cooker requires that the robot have a cloth and cleaning liquid. As well as resulting in a clean cooker it will dirty the floor. It is to be done if the stove is dirty.

ii) Cleaning the floor requires that the robot has a mop and cleaning liquid. It is to be done if the floor is dirty.

iii) Any cleaning item can be fetched from the cupboard.

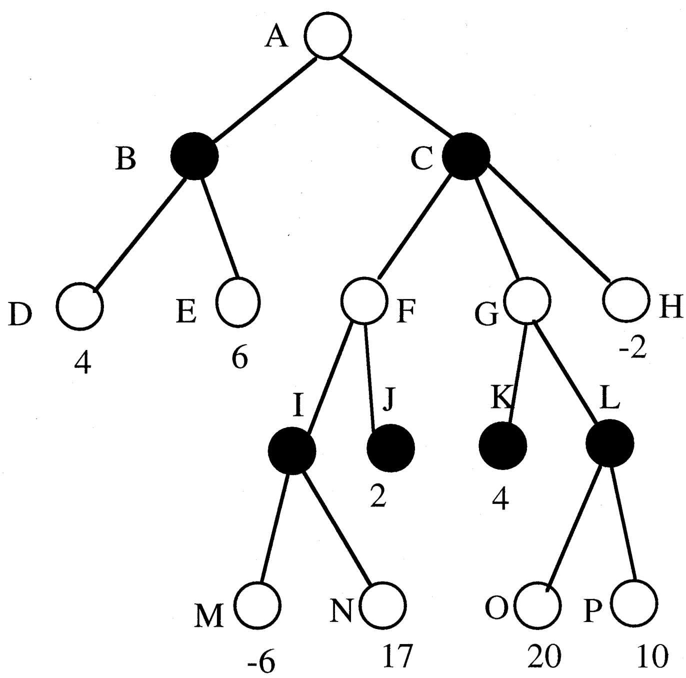
Formulate a suitable goal desciption for the robot. Give a description of an initial state that will cause the robot to plan to clean the stove.

The three parts carry respectively 40%,30%,30% of the marks.

2 a In a game playing program for two player games two heuristic values are associated with each state of the game, one negative and one positive. These are values of a heuristic function used to estimate how good the game state is from the perspective of each player, assuming it is the player's move. Briefly describe what aspects of the state of the noughts and crosses game such a heuristic function might take into account when evaluating a state of the game for the nought player, for which a good game state has a high positive value. *Do not* give an algorithm.

b i) What is the alpha value of a white (maximising) node of a game tree?
ii) What is the beta value of a black (minimising) node of a game tree?
iii) What is an alpha cut-off?
iv) What is a beta cut-off?

c Consider the following game tree in which the evaluations of the leaf nodes are as indicated. The first player is white, for whom the greater the score the better. For the second player, black, the lower the score the better.



i) Use minimax to assign backed up values to each of the nodes in the game tree. What move should white make?

ii) What nodes or subtrees will *not* be visited by the alpha-beta modification of a depth first minimax algorithm? Give the reason why each leaf node, or each subtree, will not be visited.

The three parts carry respectively 20%,30%,50% of the marks.

Turn over.....

3 a Give a brief description of binary decision tree learning. Explain how the concept of entropy can be used to guide the construction of the decision tree.

b Briefly describe the main ingredients of the production rule representation of problem solving knowledge.

c Give three production rules that will cause a working memory comprising a set of facts such as:

on(box1,box2), on(box2,box3), on(box3,floor), on(box4,box5),....

to be transformed into a set of facts such as:

on(box1,floor), on(box2,box1), on(box3,box2),...

The initial set of facts describes a world in which an unknown number of boxes are stacked, in several stacks, on the floor. The stacks contain unknown and possibly different numbers of boxes.

The final set of facts describes a world in which all the boxes are in a single stack.

The production rules should correspond to the actions of a robot arm that can move a box from the top of any stack onto the table, and can move any box B1 that is on the table so that it is on top of another box B2. This is providing neither B1 nor B2 have boxes on top of them. B2 may or may not be on top of another box. Specify the conflict resolution stragtegy that captures the behaviour of a robot arm that first unstacks all the boxes onto the floor, then builds them up into one stack.

[*Hint*: one of the production rules might fire only once and be the one that starts the construction of the single stack.]

The three parts carry respectively 40%,30%,30% of the marks.

4 Let Cl, Ns, Pr be lists of words, intended to be instances of the informal grammatical classes *clause*, *noun conjuncts*, and *predicate* respectively. Suppose that these classes combine grammatically in accordance with the Prolog rules below.
(Line numbers are included for convenient reference in your answer).

```
1    clause(Cl) :-  nouns(Ns), predicate(Pr), append(Ns,Pr,Cl).
2    nouns(Ns):- nouns (Nh), connected (Nt), append(Nh,Nt,Ns).
3    nouns ([jack]).
4    nouns ([jill]).
5    nouns ([mary]).
6    connected ([and|Ns]) :- nouns( Ns).
7    predicate(Pr) :- verb(V), adjective(A), append(V,A,Pr).
8    verb([went]).
9    verb([fell]).
10   verb([are]).
11   verb([is]).
12   adjective([up]).
13   adjective([down]).
14   append([],Z,Z).
15   append([A|X],Y,[A|Z]) :- append(X,Y,Z).
```

i) Briefly explain why a difference list formulation is to be preferred in a grammar intended for parsing a written English clause, such as the one represented by the Prolog list [mary,is,down].

ii) Show how to incorporate the use of difference lists here, illustrating with the replacements for rules 1,2,3 and 7,8, 12.

iii) Identify the source of ambiguity in parsing [jack, and, jill, and, mary] with these rules and show how to eliminate the ambiguity.

iv) Briefly describe how to then force agreement in number (singular/plural), between nouns and the verbs *are* and *is* as they may appear in a clause.

v) Briefly describe how to modify the rules to capture the content of a sentence as a Prolog database of literals expressed using predicate functors went-up, went-down, *etc.*, as in *went-up*(*john*), *is-down*(*jill*). Do not attempt to provide detailed rules.

The three parts carry, respectively, 30%, 40%, 30% of the marks.

[End of paper