UNIVERSITY OF LONDON

IMPERIAL COLLEGE OF SCIENCE, TECHNOLOGY AND MEDICINE

EXAMINATIONS 1996

BEng Honours Degree in Computing Part II
MEng Honours Degrees in Computing Part II
BSc Honours Degree in Mathematics and Computer Science Part II
MSci Honours Degree in Mathematics and Computer Science Part II
for Internal Students of the Imperial College of Science, Technology and Medicine

This paper is also taken for the relevant examinations for the
Associateship of the Royal College of Science
Associateship of the City and Guilds of London Institute

PAPER 2.11 / MC2.11

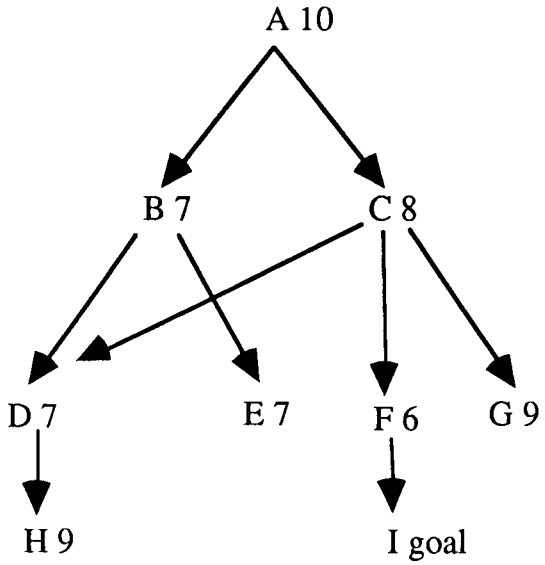
ARTIFICIAL INTELLIGENCE

Thursday, May 9th 1996, 2.00 - 3.30

Answer THREE questions

For admin. only: paper contains
4 questions
5 pages (excluding cover page)

1 In the partial search graph below, each node is labelled with a letter and a number. The number is the value of an heuristic function that tries to estimate how close the node is to a goal node. The value of this function is 0, if and only if the node is a goal node.



a For the above search graph, give the search trees that would be constructed by:

i) breadth first search
ii) best first search.

Assume that each of the search algorithms checks each newly generated node to see whether it has been generated at *any* earlier stage in the search, ignoring it if it has. On each answer search tree, number each arc with the stage in the search at which it was added to the tree.

b The following is a simplified generic search program written in Prolog. It is simplified because at each step in the search it retains only a list of the leaf nodes of the part of the search graph that it has constructed up to that point. Each leaf node contains a problem state and the value of the heuristic estimate of how far that problem state is from a goal state. The program is invoked with a query of the form:

```
search([node(start,d)],Goal)
```

where **start** is the start state for the search and **d** is the estimate of how far that start state is from a goal state. It will succeed, if at all, with `Goal` bound to a term `node(GS,0)` where `GS` is a goal state.

```
search(LeafNodes,GNode) :-
    choose(GNode,LeafNodes,RLeafNodes),
    goal(GNode).
search(LeafNodes,GNode) :-
    choose(Node,LeafNodes,RLeafNodes),
    next_nodes(Node,NextNodes),
    add_to_leaves(NextNodes,RLeafNodes,NLeafNodes),
    search(NLeafNodes,GNode).
```

Assume that the search problem is defined by a set of clauses for the relation:

`next_state(State,NextState)`: `NextState` is a state that is the result of applying one of the allowed problem solving operators to `State`

`value_of(State,V)`: `V` is an estimate of how far `State` is from a goal state for the problem

i) Give a suitable Prolog program for `next_nodes/2` and `goal/1`.

ii) Suppose that `choose/3` just selects the first node from its `LeafNodes` second argument. Give a description of what the corresponding `add_to_leaves/3` program would do (you do not need to give the program), so that the complete program implements best first search, with no loop test.

c Describe how you would modify the general `search` program given above so that it checks for duplicate occurrences of States and so never adds a State to the list of leaf nodes more than once. You do not need to give the modified program, but give the new program for `next_nodes/2` that will be needed.

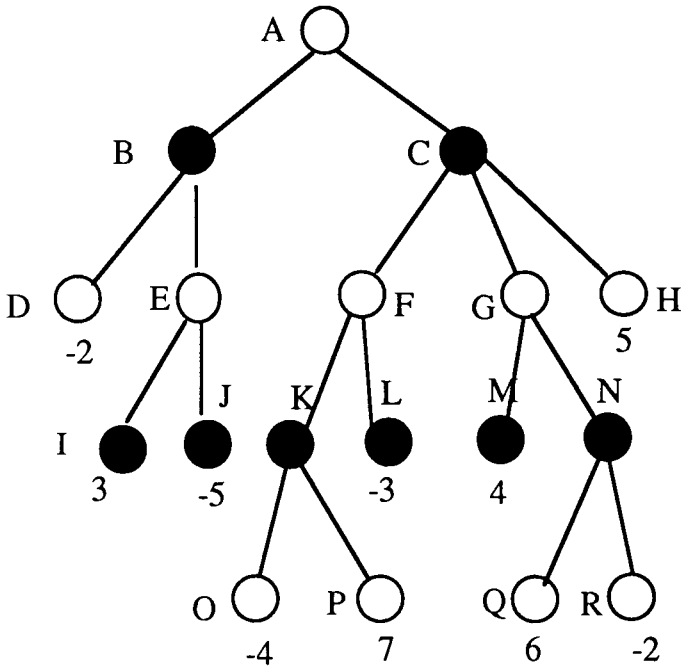
The three parts of this question carry respectively 30%, 40%, 30% of the marks.

Turn over ...

2 a In the *alpha/beta algorithm* for searching a game tree, explain and justify the pruning operations of:

i) *alpha* cut-off
ii) *beta* cut-off.

b 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 algorithm using a depth first left to right traversal of the tree? For each such leaf node, or subtree, explain why it will not be visited.

The two parts carry respectively 40%,60% of the marks.

3. Gourmet Foods Ltd wants to develop a production rule expert system that can be used in the home for advising on dishes for dinner parties depending upon the season and the dietary requirements of the guests.

The production rules are to have three phases or contexts, which progressively choose the starter, the main course and the desert . The production rules will access a 'knowledge base', supplied by the company and recorded as facts in the working memory, which has recommended dishes of various kinds, appropriate for the season (spring or summer etc) and dietary requirements. The following informally expressed production rules are to be used in the application:

for choosing a starter:
If there is no vegetarian on the guest list, then choose as a starter any of the recommended starters for the season.

If there is a vegetarian on the guest list, then choose raw vegetables as the starter.

for choosing a main course:
If the starter was raw vegetables, choose as a main course any recommended vegetarian main course for the season.

Otherwise, choose a season recommended fish main course if the starter was a meat dish, and choose a season recommended meat course if the starter was a fish dish.

for choosing the desert:
If all of the guests like fruit, and raw vegetables was not the starter, choose fresh fruit as the desert.

Otherwise, choose any season recommended desert.

a State what predicates, with what meaning, the company should use to represent the information held in the working memory of the production rules system. What predicates will have their facts supplied by Gourmet Foods Ltd, and what predicates will have facts supplied by the user of the system?

b Give production rules to implement this expert system. Assume that none of the facts to be supplied by the user are given before the exert system starts but are supplied as a result of question actions of the production rules.

You can assume the following types of actions:

add(F):	fact F is added to the working memory
ask_about(F):	user is asked to supply facts that match F. All such user supplied facts (if any) are added to working memory. The fact asked_about(F) is always added to working memory.

The two parts carry respectively 40%,60% of the marks.

Turn over ...

4.

a Briefly describe the following *frame knowledge representation* concepts, and for each one mention a related concept of object oriented programming.

i) a frame definition
ii) a frame instance
iii) an inherited value for a frame slot
iv) a slot value expression

b i) Represent the following information using some suitable frame specification notation.

A vehicle has a medium of travel, a method of propulsion and a type of cargo the defaults for which are, respectively, land, engine and people. A car is a vehicle which also has a maximum number of passengers, a manufacturer , a model name, and an engine capacity. The default maximum number of passenges is 5, the default manufacturer is 'Ford', and the default value of its engine capacity is the value of a function `default_capacity` with arguments the manufacturer of the car and the model name. Any assigned engine capacity must be at least 500. The model name N given to a car for which the manufacturer is M must be such that it satisfies the constraint relation `model_of(M,N)`. You can assume that the function `default_capacity` and the relation `model_of` are already defined. A 'Golf' is a car with maufacturer 'VW'.

ii) How many passengers can the 'Golf' carry. Justify your answer.

iii) If we tried to retrieve the engine capacity of the 'Golf' how would it be calculated?

The two parts carry respectively 40%,60% of the marks.

End of paper