COLÁISTE NA hOLLSCOILLE, CORCAIGH
UNIVERSITY COLLEGE, CORK

Summer Examinations 1999

B.Sc. Honours in Computer Science

CS486 Knowledge Based Systems

Dr. D.G. Bridge

Time allowed: **Two hours**

Answer **four out of five** questions.

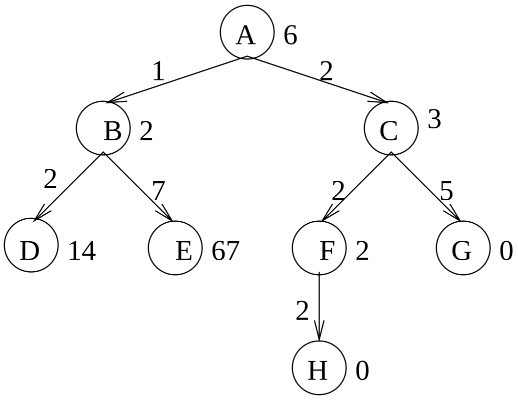
1 (10 marks)

(i) [2 marks]

(a) [1 mark] Give one advantage of *depth-first* search over *breadth-first* search.

(b) [1 mark] Give one advantage of *depth-first iterative deepening* search over *depth-first* search.

(ii) [5 marks] Consider the following *state space* in which the states are shown as nodes labeled A through H. A is the initial state and G and H are the goal states. The numbers alongside the edges represent the costs of moving between the states. To the right of every state is the estimated cost of the path from the state to the nearest goal.



Show how each of the following search strategies finds a solution in this state space by writing down, in order, the names of the nodes removed from the agenda. Assume the search halts when a goal state is removed from the agenda. (In some cases, multiple answers are possible. You need give only one such answer in each case.)

(a) Depth-first;

(b) Depth-first iterative-deepening;

(c) Least-cost search; and

(d) Heuristic search using $f(n) = g(n) + h(n)$ as the heuristic function, where $g(n)$ is the cost of the path to node n, and $h(n)$ is the estimated cost of the path from node n to the nearest goal.

(iii) [3 marks] What does it mean for an heuristic function to be *admissible?*

Is $h(n)$ of part (ii)(d) admissible?

Hence, what do we know about the search strategy of part (ii)(d)?

2 (10 marks)

The Towers of Brahma Problem

There are 3 poles, arranged in a row; there are five disks of different diameters, each having a hole in its centre so that it can be slotted onto the poles. The disks are all on the first of the poles and must be moved to the third of the poles. Only one exposed disk may be moved at a time. At all times, the disks on a pole will be in decreasing order of size (smallest on the top). When moving a disk, it can only be moved to an adjacent tower, i.e. a disk cannot be moved from the leftmost of the 3 poles to the rightmost (or vice versa) in a single move.

Your task is to formulate this problem as an AI search problem. (Extra marks will be gained for precision.)

(i) [3 marks] What representation would you use for the *states?*

(ii) [1 mark] Give the *initial state.*

(iii) [1 mark] Give the *goal state.*

(iv) [3 marks] Give *one* of the *operators.*

(v) [1 mark] How many operators would there be *in total* in your formulation?

(vi) [1 mark] Briefly explain how the other operators differ from the one you have given.

3 (10 marks)

This question uses the following 'key' for the unary predicate symbols r, h, t and s:

$$\begin{array}{lcl} r(x) & : & x \text{ is a member of the UK royal family} \\ h(x) & : & x \text{ is happy} \\ t(x) & : & x \text{ is a tax-payer} \\ s(x) & : & x \text{ is selfless} \end{array}$$

(x, x_1, x_2, x_3 and y will be used as variables.)

You are given the following *clauses*:

$$\neg s(x_1) \vee t(x_1)$$

(all selfless people pay taxes)

$$\neg t(x_2) \vee r(f(x_2))$$

$$\neg t(x_3) \vee h(f(x_3))$$

(for every tax-payer, there are happy members of the UK royal family)

(f is a Skolem function)

Use *resolution refutation* theorem-proving to show that if there's a selfless member of the UK royal family, then there's a happy member of the UK royal family, i.e. in first-order predicate calculus:

$$(\exists x(s(x) \wedge r(x))) \rightarrow (\exists y(r(y) \wedge h(y)))$$

Show your working, presenting your proof in the form of a *refutation tree*.

4 (10 marks)

The following questions concern AI planning, especially when it is carried out by *partial-order planners*, such as POP.

(i) [$1\frac{1}{2}$ marks] *"POP searches* **plan space** *rather than* **state space***."*
Explain what this means.

(ii) [$3\frac{1}{2}$ marks] *"POP is a* **regression planner***."*

(a) [1 mark] Explain what is meant by regression planning.

(b) [1 mark] Why is it generally advantageous for planners to work this way?

(c) [$1\frac{1}{2}$ marks] Describe **in detail** the part(s) of the POP algorithm that make POP a regression planner. (Do *not* describe the whole algorithm; simply describe the relevant part(s) of the algorithm.)

(iii) [2 marks] *"POP is a partial-order (or non-linear) planner, and uses a least-commitment strategy."*
Explain what this means.

(iv) [3 marks] *"POP decomposes problems and solves the resulting subproblems as independently as possible, but it looks out for, and deals with, cases where subproblems interact."*
Describe **in detail** the parts of the POP algorithm that achieve the above. (Again, do *not* describe the whole algorithm; simply describe the relevant part(s) of the algorithm.)

5 (10 marks)

(i) [8 marks] A meteorological office has collected data that it thinks can be used to predict whether it will rain tomorrow (WILLRAIN). The data records for day i whether it was raining (RAINING), whether it was cloudy (CLOUDY), whether the temperature was high (HOT) and whether Old Mother Flannery's seaweed was limp (LIMP), and it then records whether or not it rained on day $i+1$:

Record	RAINING	CLOUDY	HOT	LIMP	WILLRAIN
1	F	F	T	T	F
2	T	T	T	F	T
3	T	T	F	F	T
4	F	T	T	T	T
5	F	T	F	T	F
6	F	T	T	F	T
7	T	F	F	T	T
8	F	F	F	T	F

Assume that the data is *not* noisy.

The rule learning algorithm that was covered in lectures is being applied to this algorithm. The first rule has already been learned:

$$\text{RAINING} \rightarrow \text{WILLRAIN}$$

This covers positive examples 2, 3 and 7 and no negative examples.

Your task is to continue running the algorithm. Show **in detail** how the algorithm proceeds. Explain each step.

(ii) [2 marks] State, in general terms, how the rule learning algorithm can be modified to handle *noisy* training data.