

# UNIVERSITY OF DUBLIN

## TRINITY COLLEGE

Faculty of Engineering and Systems Sciences

Department of Computer Science

B.A.(Mod.) Computer Science  
Junior Sophister Examination

TRINITY TERM 1998

### ***3BA2 - Artificial Intelligence***

Thursday 28th May

Sports Hall

14.00 - 17.00

Dr. M. Keane, Mr. M. Brady

Attempt **FIVE** questions, at least **TWO** from each section.

Use separate answer books for each section.

### **SECTION A**

1. The representations used in symbolic Artificial Intelligence are quite different to those used in subsymbolic Connectionism. Discuss the similarities and differences between these two approaches using examples from the literature.
2. Compare and contrast an example of inductive learning with one of deductive learning and evaluate the relative strengths and weaknesses of each paradigm.
3. What should an adequate search technique do? Critically evaluate the proposition that "Intelligence is search".
4. Three learning rules -- Hebb Rule, Delta Rule and the Generalised Delta Rule -- have been used to train neural networks. Specify each rule and explain why these different learning rules are needed.

### **SECTION B**

5. Write a Prolog program to add items from a list to an ordered binary tree. Each node in the tree contains an item, a left subtree and a right subtree. The items in a node's left subtree are before (in Prolog's

- standard ordering) the node's item, and items in the right subtree are the same or after the node's item. Write a second program to generate a list, in standard order, of the items in the tree.
6. Devise a complete Prolog program to prove assertions about travelling between towns in Ireland (or another locality you are familiar with). Your program should store information about the distance by road between directly connected towns and the map coordinates of each town. Devise a cost-first program for finding a route between any of the towns. Comment on whether it discovers the shortest route. Show how you could use the map coordinates to derive an evaluation function to guide the search. Explain, in one line, what A\* search is.
  7. Outline the architectures of expert systems. Design an expert system to screen simple faults at an automated help-desk. Write the complete run time architecture of the expert system, and just give examples of the rules, as you probably won't have time to give all the rules.
  8. Discuss Kowalski's aphorism  $\text{Program} = \text{Logic} + \text{Control}$ . Give examples and explain the operation of the cut. From a strict logic point of view, give an account of the deficiencies of Prolog as a logic programming language.

© University of Dublin 1998