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 2000**

3BA2 Artificial Intelligence

Thursday 25th May Luce Hall 09.30 - 12.30

T.Fernando, P. Cunningham, M. Brady

Answer **five** questions, at least **one** from each section. Each question is worth 20 marks.

Section A

1. (a) What are the **symbol-system hypothesis** and the **Church-Turing thesis**? What do they have to do with artificial (or computational) intelligence?
   (b) What are **Turing machines**, and how do they differ from finite state machines?
   (c) Why is non-determinism a greater problem for Turing machines than for finite state machines?

2. (a) What is the **Herbrand interpretation** of a knowledge base?
   (b) What is the **Unique Names Assumption**, and do Herbrand interpretations satisfy it?
   (c) What is the **Complete Knowledge Assumption**, and do Herbrand interpretations satisfy it?

3. (a) What is the difference between **declarative** and **procedural** semantics?
   (b) What problem do the two rules

```
p :- p.
q :- q,!,fail ; true.
```

pose for declarative accounts such as the following (from *Computational Intelligence*, p.252)?

> The formula $\sim p$ means that p is false under the Complete Knowledge Assumption. This is called **negation as failure**. That is, p is false in all models of the [Clark] completion of the program. We use a different symbol to the previous negation where $\neg p$ is true in an interpretation if p is false in the interpretation, since that symbol doesn't incorporate the Complete Knowledge Assumption. Instead, we have $T \models \sim p$ iff $T' \models \neg p$ where T' is the completion of T.

4. Suppose we were to reduce an n-ary relation $r(x_1, ..., x_n)$ to the conjunction of $(n-1)$-binary relations

$$r_1(x_1, x_2),\ r_2(x_2, x_3),\ \ldots,\ r_{n-1}(x_{n-1}, x_n)$$

   (a) What is wrong with this reduction? (Hint: apply the proposal to the sentence "Ana gave Bill candy, and Dan gave Bill eggs.")

   (b) How does this compare with the reduction of an n-ary relation to semantic networks?

5. (a) What is A* search?

   (b) Give an example where A* does breadth-first search.

   (c) Give an example where A* does depth-first search.

   (d) What does it mean for A* to be **admissible**, and which (if any) of the two examples you have discussed are admissible?

6. (a) Explain the difference between supervised and unsupervised learning.

   (b) Describe the overall algorithm for building a decision tree from supervised data.

   (c) Describe how entropy can be used to determine the features to use in a decision tree.

Section B

7. Discuss Kowalski's aphorism `Program = Logic + Control`. Give examples and explain the operation of the cut. Why is the `Control` part *mandatory* rather than *optional* for a Prolog programmer.
   From a strict logic point of view, give an account of the deficiencies of Prolog as a logic programming language.

8. Give an account of how Prolog is implemented as a computer language. In your answer, explain tail recursion. Also, highlight the correspondences and distinctions that can be drawn between an implementation of Prolog and an implementation of a more conventional procedural language.