UNIVERSITY OF LONDON
IMPERIAL COLLEGE OF SCIENCE, TECHNOLOGY AND MEDICINE

EXAMINATIONS 1998

MEng Honours Degrees in Computing Part IV
MSci Honours Degree in Mathematics and Computer Science Part IV
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
Associateship of the Royal College of Science
Associateship of the City and Guilds of London Institute

PAPER 4.86

DEDUCTIVE DATABASES

Thursday, April 30th 1998, 10.00 - 12.00

Answer THREE questions

For admin. only: paper contains 4 questions

1 Each of the databases $\Delta_1, \Delta_2, \Delta_3, \ldots$ of this question is a set of sentences of first-order logic (FOL) representing a schedule of meetings and their attendees. A meeting is 'scheduled' when a record to that effect appears in the database. The predicate $c(x)$ represents that a meeting is scheduled for day x. Predicates $p(x)$, $vp(x)$, $m(x)$ are to be read as the president (resp. vice-president, manager) is available to attend a meeting on day x.

a Explain what is meant by saying that integrity constraints on 'proof theoretic' databases are often regarded as having a *metalevel* or *epistemic* reading.

Sketch how this reading can be expressed by extending the language of the database with a modal operator B ('it is believed/known that', 'it is derivable that').

b Use the language of part (a) to express each of the following as metalevel/epistemic integrity constraints:

C1: meetings should not be scheduled for days on which the president is not known to be available;

C2: meetings should not be scheduled for days on which the vice-president is known to be unavailable;

C3: a meeting should be scheduled on at least one of day 1, day 2, day 3, day 4.

c For each of the following databases, indicate which of the constraints of part (b) it satisfies and which it does not satisfy. Justify each answer briefly.

$\Delta_1 = \{\ p(1),\ \neg p(2),\ vp(2),\ \neg vp(1),\ \neg m(3),\ \ c(1) \vee c(2)\ \}$;

$\Delta_2 = \{\ p(1),\ \neg p(2),\ vp(2),\ \neg vp(1),\ \neg m(3),\ \forall x(vp(x) \rightarrow m(x)),\ c(3)\ \}$;

$\Delta_3 = \{\ p(1),\ \neg p(2),\ vp(2),\ \neg vp(1),\ \neg m(3),\ \forall x(\neg p(x) \rightarrow m(x)),\ c(3)\ \}$.

d Suppose attention is restricted to databases of the form comp(Δ)$\cup$*DCA*, where Δ is a set of ground unit clauses (or atoms), comp denotes the Clark completion, and *DCA* is the usual domain closure axiom. Explain briefly but as precisely as possible why the distinction between 'not known to be available' and 'known to be unavailable' in constraints C1 and C2 then disappears.

How can a database of this special form be implemented as a relational database?

The four parts carry equal marks.

2a Let IDB be a set of definite clauses and EDB a set of ground unit clauses. Define the closure of EDB under IDB, $Cl_{IDB}(EDB)$. (It is not necessary to define the immediate consequence operator.)

Write down the algorithms (in any style) for *naive* and *semi-naive* bottom-up computation of the 'derived tuples' $s = Cl_{IDB}(EDB) - EDB$.

b The following clauses are one way of defining the transitive closure (*path*) of a set of arcs (*arc*) :

$$path(x, y) \leftarrow arc(x, y)$$
$$path(x, y) \leftarrow arc(x, z), path(z, y)$$

Suppose that *arc* is stored in EDB. Show how the following 'differential relations' may be derived for the semi-naive evaluation of *path*:

$$\Delta PATH_1 = ARC$$
$$\Delta PATH_{n+1} = (ARC \circ \Delta PATH_n) - PATH_n$$

where $ARC = \{<x, y> \mid arc(x, y) \in EDB\}$ and $R \circ S$ denotes the composition of relations R and S, i.e.

$$R \circ S = \{<x,y> \mid \exists z\ [<x,z> \in R \ \&\ <z,y> \in S]\}$$

c The following is an alternative formulation of *path*:

$$path(x, y) \leftarrow arc(x, y)$$
$$path(x, y) \leftarrow path(x, z), path(z, y)$$

Write down the 'differential relation' formulation of semi-naive evaluation of this version of *path*. It is not necessary to expain the derivation in detail.

d Show that magic set transformation of the *path* clauses in part (b) for the query *path(a, y)* ? yields the following set of clauses, when followed by a suitable renaming of predicates.

$$path(x, y) \leftarrow relevant\_arc(x, y),$$
$$path(x, y) \leftarrow relevant\_arc(x, z), path(z, y)$$

$$relevant\text{-}arc(x, y) \leftarrow arc(x, y), req\_path(x)$$

$$req\_path(z) \leftarrow relevant\_arc(x, z)$$

$$req\_path(a)$$

The four parts carry equal marks.

Turn over ...

3 For each of the databases IDB$\cup$EDB in parts (a), (b), (c), compare the following semantics for the contents of the database. In each case indicate which answers (if any) are correct for queries s ? and w ?.

i) logical consequences of the Clark completion, comp(IDB$\cup$EDB);

ii) the closure(s) of 'facts' EDB under 'rules' IDB, or equivalently, the minimal Herbrand model(s) of IDB$\cup$EDB (not necessarily unique);

iii) the Apt-Blair-Walker ('iterated fixpoint') semantics;

iv) the stable model ('answer set') semantics.

It is *not* necessary to give definitions or proofs for standard results but you should justify your answer carefully.

a EDB = $\{f, g\}$, IDB $= \{s \leftarrow f, g, \neg w\ ,\ w \leftarrow \neg g\ \}$

b EDB = $\{f, g\}$, IDB $= \{s \leftarrow f, g, \neg w\ ,\ w \leftarrow g, \neg s\ \}$

c EDB = $\{f, g\}$, IDB $= \{s \leftarrow f, g, w\ ,\ w \leftarrow g, \neg s\ \}$

The three parts carry equal marks.

4a In Reiter's default logic, what are *normal* and *non-normal* default rules?

Define the *extension* of a default theory $<D, W>$.

b Formulate the following as default rules:

Europeans are assumed civilised, unless they are known to be drunk.

Football supporters are assumed drunk, unless they are known to be educated.

There are several possible readings for these natural language statements: choose one. What conclusion does your formulation reach in the case of an educated European football supporter?

c In autoepistemic logic, what is a *stable expansion*?

State, without proof, how Reiter default theories can be translated into autoepistemic logic.

d Explain how the stable model ('answer set') semantics of normal logic programs can be regarded as a special case of autoepistemic logic. Concentrate on the main ideas. It is not necessary to give formal proofs of all steps.

The four parts carry equal marks.

End of paper