# UNIVERSITY OF DUBLIN CS4B1A1

## TRINITY COLLEGE

FACULTY OF ENGINEERING & SYSTEMS SCIENCES

DEPARTMENT OF COMPUTER SCIENCE

**B.A. (Mod.) Computer Science** **Trinity Term 1999**
**Degree Examination**

4BA1 INFORMATION SYSTEMS

Monday, 31st May 1999 Drawing Office 14.00–17.00

Mr. V. Wade and Dr. M. Mac an Airchinnigh

Attempt **five** questions, at least **two** from each section.
Please use separate answer books for each section.
Students may avail of the HANDBOOK OF MATHEMATICS of Computer Science

## SECTION A

**1.**

A Library information system contains a database concerning books, publishers, authors, library branches, and library borrowers. For every book the library maintains a book number and a catalog number. Every copy of a book has a unique book number and copies of the same book share the same catalog number. (This facilitates the library maintaining many copies of the same book!).

Also maintained for each book is the book title, publisher name, publisher address and publisher phone number. Book titles are not unique. An author can be the author of several books and a book can have several authors. One publisher can publish many books, but each publisher name is unique. For each library branch, the branch name and address are maintained. Also maintained for each library branch is the number of copies of each book.

\ . . . continued over.

Every borrower is identified by a library card which has a unique number within a particular library branch. The library stores information about the borrower name, address and phone number. Also identified are the books that each borrower currently has on loan, the date the book was borrowed and the return date for the book. The library has a policy of not allowing borrowers to take out more than five books at any time.

(a) Draw a Functional Dependency diagram for the relations in the above database stating any assumptions made in your solution.

(b) Derive a set of fully normalised relations from the determinancy diagram, indicating clearly the primary key (candidate identifier(s)) and foreign keys of each relation. Also identify constraints (table or assertions) which you may deem important based on the above library description.

(c) Develop SQL statement(s) to do the following and comment on each of your answers with respect to integrity constraints.

(i) Create a view which, for all branches, identifies the borrowers and the books they borrowed.

(ii) Check if a borrower can borrow a book entitled 'The Wind in the Willows.' (you may assume the borrower's card details are available to you as you are writing your SQL and that the library branch number in question is 1234.

(iii) Delete a copy of a book (book number 1559) from a branch of the library.

(iv) Perform some market research to identify the Authors who are of interest to borrowers with an address in 'DUBLIN.' You may assume that the city name forms part of the borrowers address.

**2.**

(a) Compare and contrast (i) two phase locking which uses a deadlock prevention protocol (which you have studied) AND (ii) (pure) timestamping as algorithms for achieving concurrency control. In your answer outline the algorithms for two phase locking using a deadlock prevention protocol of your choice AND the algorithms involved in realising pure timestamping concurrency control.

(b) Briefly comment on two phase locking with deadlock prevention & pure timestamping with regard to

(i) levels of concurrency achieved under varying transaction mixes;

(ii) granularity of resource concurrency.

3.

(a) What are Integrity Constraints? Give examples of three types of integrity constraint.

(b) 'Violation of integrity constraints' is one of the 3 well known types of transaction interference, which can occur in concurrently executing transactions. Explain what this is and how it can arise, illustrating your explanation with an example.

(c) What is Serialisability and what is it used for? Outline an algorithm, which can check a transaction schedule and determine if that is serialisable, using read/write and write/write conflicts.

(You may use pseudocode to describe your algorithm and you may assume that a transaction schedule is input <u>in a form</u> similar to the example below:

$$S = R_1(x), R_2(y), R_3(z), W_3(z), R_3(x), W_1(x)$$

where $R$ indicates a Read operation, $W$ indicates a Write operation, subscripts indicate the transaction number and the letter in round brackets indicates the resource on which the operation is to occur.)

4. A County Council Dwellers Database stores information about people for whom they are currently providing accommodation. The system maintains information about a Dweller's name (non-unique), description (textual), date of birth and social security number (unique). The system also contains details of the dwellings themselves e.g. dwelling address, number of bedrooms, floor area of dwelling, dwelling type information. There are several types of dwelling type which are based on a combination of house type (apartment, two storey house, bungalow) and attachment (terraced, semi-detached, detached). The system also tracks which dwellers are currently habitants of which residence and the rent being applied to each residence. The rent for a particular dweller can be calculated by dividing the rent of the residence by the number of people currently residing there. The default rent for all dwellings is IEP 1000. Note: the Council ALLOWS an individual to rent more than one dwelling.

(a) Give the functional dependency diagram for the entities represented in this database, stating any assumptions made in your answer.

(b) Derive a relational schema for the database which is in Boyce/Codd normal form.

(c) Write the SQL statement required to create this database, including any constraints you deem appropriate.

(d) Write SQL statements to do the following:

(i) List the names of the dwellers who live alone.

(ii) Create a view of the dweller names, addresses and residence information.

(iii) Delete all residences where the rent is IEP 1000.

# SECTION B

**5.** Consider the following fundamental principle:

> *With a view to requirements modelling, the purpose of a model is to ask questions and demonstrate that answers can be given entirely in terms of the model. If such answers can not be found then the model is inadequate.*

Entities such as corporations, families, and individual persons, etc., may *subscribe* to a collection of telecommunications services ($X \in \mathcal{P}SRV$) and are considered as elements of the space of subscribers $s \in SUB$. The employees of a corporation are the actual end-users ($e \in USR$) of the services. We may formally specify these relationships between subscribers, end-users and services by the model:

$$\mu \in SUB \to (USR \to \mathcal{P}SRV) \tag{1}$$

The element $\mu$ is said to constitute the system at any particular moment in time.

(a) With respect to the model (1), demonstrate that each of the following questions may be answered appropriately by writing down formal mathematical expressions, with accompanying pre-conditions where necessary.

(i) Which services are used by the employee $e$ of subscriber $s$ in the system $\mu$?

(ii) What is the effect on the system $\mu$, of a new employee $e$ who joins the corporation $s$ but has not yet been given access to any services?

(iii) What is the effect on the system $\mu$, of an employee $e$ who leaves the corporation $s$?

(b) Service providers ($p \in PRV$) keep a record, both of the services that they offer on a network:

$$\delta \in PRV \to \mathcal{P}SRV \tag{2}$$

and, in addition, of the subscribers who pay for them:

$$\kappa \in PRV \to (SUB \to \mathcal{P}SRV) \tag{3}$$

How can we be sure that the services provided by $p$ in model (2) are exactly those recorded in model (3)?

(c) Who are the end-users of the services provided by $p \in PRV$?

**6.** Let us consider some models of a computing environment:

$$\begin{aligned}
\alpha &\in ENV_0 = VAR \to VAL \\
(\tau, \mu) &\in ENV_1 = (VAR \to LOC) \times (LOC \to VAL) \\
\gamma &\in ENV_2 = ZONE \to (VAR \to VAL)
\end{aligned}$$

where *VAR*, *VAL*, *LOC*, and *ZONE* denote sets of variables, values, locations, and computation zones, respectively. Both $ENV_1$ and $ENV_2$ are considered to be elaborations of $ENV_0$ which is the basic abstract model.

(a) Show how to obtain retrieve functions $ENV_1 \xrightarrow{\Re_1} ENV_0$ and $ENV_2 \xrightarrow{\Re_2} ENV_0$ and indicate any necessary constraints or invariants on the models that you deem necessary.

(b) The semantics of assignment may be tersely represented by the mathematical expressions

$$A_0[x,a]\alpha := \alpha \dagger [x \mapsto a]$$
$$A_1[x,a](\tau,\mu) := (\tau, \mu \dagger [\tau(x) \mapsto a])$$
$$A_2[z,x,a]\gamma := \gamma \dagger [z \mapsto \gamma(z) \dagger [x \mapsto a]]$$

Prove that $A_1[x,a]$ is a correct elaboration of $A_0[x,a]$, subject to appropriate conditions which you will identify.

(c) Let us define a zone $z$ to be an element of the (finite) field $\mathbb{Z}_p$, where $p$ is prime:

$$ZONE = \mathbb{Z}_p$$

Consider a section $ENV_0 \xrightarrow{\Gamma} (ENV_0 \times ZONE)$, $\Gamma(\alpha) = (\alpha, \zeta(\alpha))$, of the (trivial) fibre bundle

$$ZONE \longrightarrow ENV_0 \times ZONE \xrightarrow{\pi} ENV_0$$

where $\zeta\colon ENV_0 \longrightarrow ZONE$ is a distribution function which assigns computing environments to zones. Let $S = \mathsf{rng}\,\gamma$. Then we may consider an element $\gamma \in ENV_2$ to be formally equivalent to the restricted section $\lhd_S \Gamma = \{\alpha \mapsto \Gamma(\alpha) \mid \alpha \in S\}$. Explain clearly the relationship between the addition of a new fibre $([x \mapsto a], \zeta([x \mapsto a]))$ to the section $\lhd_S \Gamma$ and the fundamental indexed monoid $(ENV_2, \dagger, \theta)$.

7. The $\Sigma^\star$-morphisms $\psi$ from the free monoid over an alphabet $\Sigma = \{a_1, a_2, \ldots, a_n\}$, denoted $(\Sigma^\star, \cdot, 1)$, into a monoid $(M, +, e)$ form a collection of fundamental word processing algorithms and, by extension, list processing algorithms.

(a) In general, for computability, the morphism $\psi$ is defined constructively by

$$\psi(\langle a \rangle \cdot w) = F(a) + \psi(w)$$

where $a \in \Sigma$, $w \in \Sigma^\star$ and $F\colon \Sigma \to M$ is a total function on the alphabet $\Sigma$. Show that for an arbitrary word $w$, $\psi(w)$ may be written in the closed form of a reduction:

$$\psi(w) = {}^{+}/ \circ F^\star w$$
$$\psi(1) = e$$

(b) Let $\psi_w$, $w \in \Sigma^\star$ denote the tail-recursive function which corresponds to the $\Sigma^\star$-morphism $\psi$. Show that $\psi_w$ has the closed form:

$$\psi_w(m) = m + {}^+/ \circ F^\star w$$
$$\psi_1(m) = m$$

(c) A monoid $(H, \otimes, 1)$ is said to act on a monoid $(K, \oplus, 0)$ if

$$\begin{aligned}(h_1 \otimes h_2)k &= h_1(h_2 k)\\ 1k &= k\end{aligned}$$

Let $\Psi$ denote the space of tail-recursive forms $\psi_w$, $w \in \Sigma^\star$. Show that $(\Psi, \otimes, \psi_1)$ acts on $(M, +, e)$.

**8.** A spelling-checker dictionary $\delta$ is simply modelled as a set of words taken from some set *WORD*. It is customary to regard it as an element of $\mathcal{P}$*WORD*. In the category **Sets** such elements are called the *points* of the set $\mathcal{P}$*WORD* and are denoted by maps of the form $\mathbf{1} \xrightarrow{\delta} \mathcal{P}WORD$ where $\mathbf{1}$ denotes the terminal object in the category **Sets**. The space of all such dictionaries may be modelled by the poset category $(\mathcal{P}WORD, \subseteq)$ which we shall denote by **Dict** for convenience.

(a) The specification of a *new* dictionary is given by

$$N() := \emptyset$$

The new dictionary object $\emptyset$ is a *point* of the set $\mathcal{P}$*WORD* in the category **Sets**. How does this compare with the new dictionary object in the category **Dict**?

(b) What dictionary specifications correspond to product and coproduct (i.e., sum) objects in the category **Dict**? Give details.

(c) There is a standard poset representation construction which can be transferred to poset category representation. Specifically, for each $a$ in the poset $(X, \preceq)$ define the set $S_a$ by

$$S_a := \{x \mid x \in X, a \preceq x\}$$

This construction defines a mapping $X \xrightarrow{f} \{S_a \mid a \in X\}$ and the resulting collection of sets forms a poset under set inclusion, called the poset of upward closures of $X$.

Let $\mathbf{Dict} \xrightarrow{\mathcal{F}} \mathbf{UpCl}$ be the poset category functor which corresponds to the mapping $f$. Show with the aid of a simple example that there is an associated functor $\mathbf{Dict} \xleftarrow{\mathcal{G}} \mathbf{UpCl}$ such that the pair $(\mathcal{F}, \mathcal{G})$ forms a Galois correspondence.