COMPUTER SCIENCE TRIPOS Part IB

Thursday 2 June 1994 1.30 to 4.30

Paper 6

Answer **five** *questions.*
No more than **two** *questions from any one section are to be answered.*
Submit the answers in five **separate** *bundles each with its own cover sheet.*
Write on **one** *side of the paper only.*

SECTION A

1 Introduction to Computer Architecture

The MIPS R2000 architecture allows for up to four external coprocessors. Our local systems are configured with one: the R2010 Floating Point Accelerator (FPA).

Explain how the MIPS external coprocessors are used by describing, in architectural terms, what happens when the instructions below are executed. Details specific to floating point are *not* required.

```
lwc1     $f4, fdata         # $f4 is a floating point register
                            # fdata labels a data area
mtc1     $f6, $16           # $16 is a general purpose register
mul.s    $f4, $f4, $f2

mfc1     $f4, $17

add.s    $f4, $f4, $f6
swc1     $f4, fdata+4
```

[12 marks]

Give an example of an exception which is detected by the R2000 but concerns the R2010. [2 marks]

Give an example of an exception which is detected by the R2010. [2 marks]

What provision has been made for these exceptions to be signalled between the main processor and coprocessor? [4 marks]

2 Computer Structures

Write short notes explaining the following:

(*a*) hard-wired control of a CPU [6 marks]

(*b*) asynchronous operation of a bus [7 marks]

(*c*) delayed branching in a pipelined RISC processor [7 marks]

3 Digital Communication I

Define the term *flow control*. [5 marks]

How does it differ from *congestion control*? [3 marks]

What is meant by the terms *entry level*, *hop by hop* and *end to end* flow control? When is each appropriate? [8 marks]

Sketch the design of a simple flow control protocol. [4 marks]

4 Graphics

Discuss transformations applied to 3D wireframe objects. [12 marks]

Discuss the use of homogeneous coordinate representations

(*a*) for presenting concepts [4 marks]

(*b*) within programs [4 marks]

SECTION B

5 Programming in C

Write a program in C which can solve cryptarithmetic puzzles in the format of the sum of two words. For example, given the input

```
 SEND
+MORE
-----
MONEY
```

the program would output

```
 9567
+1085
-----
10652
```

N.B. Each letter represents a different digit. [20 marks]

6 Programming Language Compilation

Discuss two possible strategies that you might use to translate the abstract syntax tree corresponding to an integer expression composed of simple variables, integer constants and the usual integer operators +, −, ∗ and / into reasonable quality code for a machine with eight general-purpose registers.

You should pay particular attention to how you would control the allocation of registers and anonymous store locations, and you should outline what optimisations are convenient to perform. [20 marks]

7 Concurrent Systems

A transaction processing system for a banking application is to be implemented using object semantics. Bank account objects include the following among their operations:

```
credit (account_id, amount)
debit (account_id, amount)
add_interest (account_id)
```

By defining "transactions" based on this example, show what is meant by

(*a*) a non-serialisable execution schedule [3 marks]

(*b*) a non-strict execution schedule leading to a cascading abort [3 marks]

Explain the ACID properties of transactions, drawing on the above application for examples. Indicate which properties are concerned with failure resilience and which with concurrency control. [8 marks]

Again using the above application for examples, explain concurrency control based on

(*a*) two-phase locking [3 marks]

(*b*) time-stamp ordering [3 marks]

8 Databases

Describe how a data model is represented in a *relational database*, and explain how one might specify a *relational database schema*. [5 marks]

What is meant by a *referential integrity constraint* in a relational database? [3 marks]

Each year the number of tourists coming to Cambridge increases by 10%. Most of the pressure falls on a limited number of identified sites in the city centre. The Tourist Board has restricted the size of any group visiting such a site to 20, and requires a group of ten people or more to get a permit in advance. Most bookings are made either by tour operators or directly by independent guides: the Tourist Board will arrange guides for groups if asked to do so.

A database is being installed to coordinate bookings and to provide information about the opening times of sites. Each site has separate opening times for summer and winter (owing to college autonomy, changes of season differ from site to site). Permits are issued to start on the hour or on the half-hour: they are valid either for 1 hour or for 2 hours, the duration being fixed for each site. The final permits of each day are timed to expire at the site's closing time. Each site has a fixed capacity, and no booking can be accepted that would cause it to be exceeded. The charge for a permit depends only on the site and the season. (Occasionally sites are closed for several hours during the normal opening period, for example when recording is taking place in King's College Chapel. The protocol is to inform the Tourist Board at least 6 months in advance.)

The Tourist Board issues permits to visit an identified site at a given time on a given day, specifying the booking agent and the number in the group. Bookings can be made up to 6 months beforehand. Permits are issued to registered tour operators and guides on account, but in all other cases payment must be made in advance. The data held for registered guides includes not only account details but also their working hours and charges.

Design a schema for the relational database that is to record this information for the Tourist Board. You may find it helpful to use domain types `DATE`, `TIME` and `MONEY` in addition to standard programming language datatypes. You do not need to specify the transactions that maintain the database, but you should state clearly any assumptions that influence the schema design. [12 marks]

SECTION C

9 Foundations of Logic Programming

Describe in detail an algorithm for finding the most general unifier of two terms. Illustrate your answer by unifying the following pairs of terms:

$$\begin{array}{rcl} f(x,a,x) & \text{with} & f(a,y,b) \\ f(x,y,z) & \text{with} & f(g(y),z,a) \\ f(g(y),y,z) & \text{with} & f(x,z,x) \end{array}$$

The variables above are x, y and z. [8 marks]

"The resolution method relies on *most general* unifiers because they are unique." Discuss. [3 marks]

The resolution method can be applied directly to any first-order formula, regardless of its structure. Discuss and evaluate the following proposals for dealing with special cases:

(a) If the formula has the form $\neg A$, then apply the resolution method to A. Failure to prove A establishes that $\neg A$ is a theorem.

(b) If the formula has the form of a disjunction $A \vee B$, then apply the resolution method separately to A and to B. If either proof succeeds then $A \vee B$ is a theorem.

(c) If the formula has the form of a conjunction $A \wedge B$, then apply the resolution method separately to A and to B. If both proofs succeed then $A \wedge B$ is a theorem.

(d) If the formula has the form $A \to B$, convert A to clauses. Then apply the resolution method to B, allowing A's clauses to take part in applications of the resolution rule. If this proof succeeds then $A \to B$ is a theorem.

[9 marks]

10 Foundations of Functional Programming

Describe the operation of a graph reducer and its treatment of the combinators **K**, **S**, **Y**, **if** (for conditional expressions) and **mult** (integer multiplication). [6 marks]

Describe the operation of the SECD machine, including its treatment of recursive functions. [5 marks]

Exhibit an infinite family Φ_n of distinct fixed-point combinators. Justify your answer by showing that $\Phi_n \twoheadrightarrow F(\Phi_n F)$ for all non-negative integers n and λ-terms F. You must also show that $\Phi_m \neq \Phi_n$ for $m \neq n$, quoting standard results about the λ-calculus if necessary. [9 marks]

11 Computation Theory

Explain *Turing's Thesis.* [5 marks]

(a) What is meant by saying that a Turing machine has *searching* states? Show that any Turing machine computation can be effected by a machine with searching states, equivalent in the sense that the head movements are identical and the same symbols are written to the tape. [5 marks]

(b) Show that, subject to suitable encoding, any computation can be carried out by a Turing machine having only two states. [10 marks]

12 Software Engineering

Compare and contrast the relative merits of Z and VDM as tools for specifying and developing large software systems. [20 marks]