# COMPUTER SCIENCE TRIPOS Part Ib

Tuesday 6 June 1995 1.30 to 4.30

Paper 4

*Answer* **five** *questions.*
*Submit the answers in five* **separate** *bundles each with its own cover sheet.*
*Write on* **one** *side of the paper only.*

## 1 Further Modula-3

Describe the facilities provided in Modula-3 for abstraction through object inheritance and partial revelations of opaque types. [8 marks]

Illustrate your answer by reference to the Trestle toolkit, mentioning the relationship between `VBT.T`, `VBT.Leaf`, `VBT.Split`, `Filter.T` and their sub-types. [4 marks]

Explain how overriding methods such as mouse, position and keyboard in a sub-type of `VBT.T` can be used to define a new class of window. [4 marks]

How is initialisation of super-types handled? [4 marks]

## 2 Concurrent Systems

An expansion card to manage a number of telephone lines is to be added to a workstation. The hardware is able to pick up and put down lines, dial a number and answer the phone. It generates interrupts to signal conditions such as a phone line being answered.

(*a*) What additions have to be made to the operating system to allow programs to use the phone hardware? [4 marks]

(*b*) Suggest a mechanism by which the operating system can block a user process until a phone line is answered. [4 marks]

A *voice mail server* is to be developed and given exclusive use of the lines. It is to be activated by a user clicking the button on his/her active badge, a mobile device allowing a user's location to be determined. The voice mail server then phones the user at the phone nearest to him/her. After authentication by typing a code, the server reads the user's mail to him/her using a speech synthesiser. The pseudo-code procedure shown below is part of this service. Every time a badge click is detected, a thread is forked to execute procedure `ActiveBadge_Click_Handler`.

```
VAR line : ARRAY [1..NUM_LINES] OF {Free,Busy};

PROCEDURE ActiveBadge_Click_Handler(user: STRING;
                                    location: STRING);
VAR which_line: INTEGER;
BEGIN
  which_line := 1;
  WHILE (line[which_line] = Busy) DO  (* Loop until a line *)
    which_line := which_line+1;       (* becomes free *)
    IF (which_line > NUM_LINES) THEN which_line := 1;
  END;
  line[which_line] := Busy;           (* Mark line as busy *)
  Pickup(which_line);
  Dial_Authenticate_Speak(which_line,user,location);
                                      (* Does clever stuff *)
  Hangup(which_line);
  line[which_line] := Free;           (* Mark line as free *)
END;
```

(*c*) Explain whether or not it is possible to implement the server using threads provided by a run-time system. [4 marks]

(*d*) Outline a possible race condition in the program. On what types of system could it occur? [4 marks]

(*e*) How could the program be amended to avoid the race condition? Note: Language-specific details are unimportant. [4 marks]

## 3 Regular Languages and Finite Automata

Define a *regular grammar*. [2 marks]

Define a *regular expression*. [2 marks]

Show that regular grammars and regular expressions characterise the same class of languages. [6 marks]

The syntax of a propositional calculus can be described by the context-free grammar $G = \langle Vn, Vt, P, S \rangle$

$$\begin{aligned} &\text{where } Vn = \{S\}, Vt = \{not, if, then, and, or, p, q, r\}, \\ &P = \{S \to p,\ S \to q,\ S \to r, \\ &\quad S \to not\ S, \\ &\quad S \to if\ S\ then\ S, \\ &\quad S \to S\ or\ S, \\ &\quad S \to S\ and\ S\} \end{aligned}$$

Construct a push-down automaton which accepts the set of strings generated by $G$. [10 marks]

## 4 Compiler Construction

Outline how minimum cost code can be compiled from a parse tree using rules that consist of tree template replacements, costs and corresponding code. Illustrate your answer by considering the translation of the expression

```
Add(Add(K1, K2), Add(K3, K4))
```

using the following rules:

```
#1  Ri  <-  Add(Ri, Rj)            cost: 2   code: ADDR  Ri,Rj

#2  Ri  <-  Add(Ri, Kc)            cost: 3   code: ADDI  Ri,c

#3  Ri  <-  Kc                     cost: 2   code: LOADI Ri,c

#4  Ri  <-  Add(Ri, Add(Rj, Kc))   cost: 4   code: ADDRI Ri,Rj,c
```

In your answer you should derive the finite state machine needed for the efficient matching of these four rules, and you should also give the cost and resulting translation of the given example expression. [15 marks]

Briefly discuss in what ways this algorithm may fail to generate optimum code when used in a compiler. [5 marks]

## 5 Data Structures and Algorithms

What is a *priority queue*? Explain the data structure known as a *heap* and document how a heap is stored in a simple linear block of memory. [4 marks]

If a heap stores $N$ items, describe how it can be viewed as an almost-balanced binary tree. What difference can there be between the greatest and least lengths of paths from the root of the tree to a leaf? What operations must be performed to move from one node in the tree to ($a$) its parent and ($b$) its offspring? [5 marks]

Describe, and estimate the costs of, procedures to

($a$) insert a new item into an existing heap;

($b$) delete the topmost item from a non-empty heap;

($c$) starting from an array holding $N$ items in arbitrary order, rearrange those items so that they form a heap, taking time less than that which would be needed if the items were just inserted into the heap one after the other. [6 marks]

A *stable* sorting method is one where items whose keys compare as equal will appear in the output in the same order that they appeared in the input list. Would a heap sort based on the algorithms you have documented be stable? Justify your answer. [5 marks]

## 6 Structured Hardware Design

Describe a *test vector* and how a set of such vectors can be used in conjunction with a digital system tester to test a hardware module. [7 marks]

Give a Verilog language description of a circuit which has three inputs (including the clock), two outputs and which contains at least one flip-flop and at least one two-input gate. [4 marks]

Give a set of test vectors which yield 100% fault coverage for the module that you defined in the previous section. [4 marks]

Explain the need for special test modes and pins in certain hardware structures. What are the advantages and disadvantages of test inputs compared with test outputs? [5 marks]

## 7 Operating System Functions

An operating system for a *smart card* is required — that is a common run-time system for the smart card, which hosts some set of applications loaded into the card at the time the card is issued.

What operations should the memory management component of the operating system support in such a system? [12 marks]

The hardware engineers are concerned about complex memory management hardware consuming too much power, but the customer demands memory protection between the applications. Describe some solutions that could meet these demands. [8 marks]

## 8 Graphics

Explain the purpose and operation of the A-buffer in rendering a sequence of images into a framestore. [12 marks]

Exhibit an example that shows an advantage over the use of a Z-buffer. [8 marks]

## 9 Computation Theory

Explain what is meant by a *primitive recursive* function and by a *partial recursive* function. [6 marks]

Show that the function giving the next state of a register machine in terms of the current state is primitive recursive. (You may assume the existence of primitive recursive functions for coding any $n$-element list of numbers $(x_1, \ldots, x_n)$ as a number $[x_1, \ldots, x_n]$ (for each $n$), and for extracting the head $x_1$ and the (coded) tail $[x_2, \ldots, x_n]$ from such a coded list.) [8 marks]

Deduce that every register machine computable partial function is partial recursive. [5 marks]

Is the converse true? [1 mark]

## 10 Numerical Analysis I

The Newton–Raphson formula for solution of $f(x) = 0$ is

$$\hat{x} = x - \frac{f(x)}{f'(x)}.$$

By means of a sketch graph, describe how the method works in a simple case. [4 marks]

When the method converges, what *rate of convergence* is expected? Describe one circumstance in which the method may fail to converge. [4 marks]

Consider the simultaneous equations

$$\left.\begin{aligned} f_1(x_1, x_2) &= x_2 - {x_1}^2 - 2 = 0 \\ f_2(x_1, x_2) &= x_1(x_2 - 3x_1) = 0 \end{aligned}\right\} \tag{1}$$

Suppose the iterative scheme

$$\begin{pmatrix} -2x_1 & 1 \\ x_2 - 6x_1 & x_1 \end{pmatrix} \begin{pmatrix} h_1 \\ h_2 \end{pmatrix} = \begin{pmatrix} -f_1(x_1, x_2) \\ -f_2(x_1, x_2) \end{pmatrix} \tag{2}$$

is applied to the equations (1). If $\{x_1, x_2\}$ is the starting approximation, the improved approximation is given by

$$\begin{pmatrix} \hat{x}_1 \\ \hat{x}_2 \end{pmatrix} = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} + \begin{pmatrix} h_1 \\ h_2 \end{pmatrix}.$$

Suppose $x_1 = 0$. Show, by solving the equations (2) that the first iteration always produces the same improved approximation for any non-zero $x_2$. [10 marks]

Verify that the method converges if $x_1$ is set to 0, and $x_2 \neq 0$. [2 marks]