COMPUTER SCIENCE TRIPOS Part IA

Monday 2 June 1997 1.30 to 4.30

Paper 1

Answer **two** *questions from Section A, and* **one** *question from* **each** *of Sections B, C, D and E.*

Submit the answers in six **separate** *bundles, each with its own cover sheet. On each cover sheet, write the numbers of* **all** *attempted questions, and circle the number of the question attached.*

Write on **one** *side of the paper only.*

SECTION A

1 Foundations of Computer Science

The datatype `PRI` defined below is to be used for the representation of priority queues which are finite or infinite ordered sets of integers.

```
datatype PRI = E
             | N of int*(unit->PRI);
```

Define an ML function `intfromto(i,j) : (int*int)->PRI` which will return a representation of the ordered set of integers

```
{ i, i+1, ..., j }
```

Define the function `first(p) : PRI->int` that will return the first (and hence smallest) integer in the given queue `p`, and `rest(p) : PRI->PRI` that will return (if possible) a representation of the given queue `p` with its smallest element removed. Your implementation should be such that the expression

```
first(rest (intsfromto(20, 1000000)))
```

should evaluate efficiently.

Define an ML function `ins(i,p) : (int*PRI)->PRI` which will return a priority queue with the integer `i` inserted in the proper position of the given queue `p`.

[10 marks]

2 Discrete Mathematics

Suppose set S has m elements and set T has n elements. Give explicit formulae involving m and n only for the functions

$$f(m,n) = |B|/|I| \qquad \text{and} \qquad g(m,n) = |B|/|S \to T|$$

where B and I are the subsets of $S \to T$ consisting of those functions which are respectively bijections and injections.

[Hint: for f it may help to consider for each subset $X \subseteq T$, the number of injections which have range exactly X and then count the number of such X.] [10 marks]

3 Programming in Modula-3

Consider the following program:

```
MODULE Squares EXPORTS Main;

IMPORT IO;

PROCEDURE Square (i: INTEGER): INTEGER =
  BEGIN
    RETURN i * i;
  END Square;

BEGIN
  FOR i := 1 TO 10 DO
    IO.Put (Fmt.Int (i) & "^2 = " & Fmt.Int (Square (i)) &
                                                  "\n");
  END;
END Squares.
```

Rewrite the program to use an exception to return the result of the function.
[10 marks]

4 Operating Systems

Describe the various functions involved in interrupt handling. Indicate the hardware and software that might be involved in their implementation. [7 marks]

Discuss the interaction of interrupt-driven software and process scheduling in an operating system. [3 marks]

SECTION B

5 Foundations of Computer Science

Noughts and Crosses is a game played by two players (O and X) on a board with nine positions numbered as follows:

1	2	3
4	5	6
7	8	9

The players place their marks (O and X) in unoccupied positions on the board until the game is complete. A completed game is when either

(*a*) there is a straight line of three Xs giving a win for X, *or*

(*b*) there is a straight line of three Os giving a win for O, *or*

(*c*) all nine positions are occupied, in which case the game is drawn.

O is the first player to move.

It is required to construct an ML structure representing the tree of all possible games. Each node of the tree should represent a reachable board state, with the root being the empty board, and the leaf nodes corresponding to won, lost or drawn games.

Define the ML datatype `tree` that you would use to represent this game tree.

[3 marks]

Define the function `mktree : unit->tree` to construct the complete game tree, explaining carefully how it works. There is no need for your implementation to be efficient in either space or time. [10 marks]

Briefly discuss ways in which your implementation of `mktree` could be made more efficient. [4 marks]

Define a function `Owins : tree->int` which when applied to the complete tree will yield the number of distinct games in which O wins. [3 marks]

6 Foundations of Computer Science

A rooted directed graph has vertices identified by integers. Each vertex `v` has a left successor given by `left(v)` and a right successor given by `right(v)`, where `left` and `right` are ML functions of type `int->int`. The graph contains the root and all vertices reachable by paths from the root.

Define a datatype `G` that could be used to represent such a graph with given root, and left and right functions, and define a function `mkgraph(root, left, right)` that can create values of type `G`. Show that such values can be used to represent both finite and infinite graphs. [4 marks]

A path through the graph is represented by a `bool list` with `true` and `false` indicating left and right edges, respectively.

Define the function `last : G -> bool list -> int` that will yield, for a given graph, the identity of the vertex reached by following the given path from the root. [3 marks]

In a new application, where `last` is repeatedly called, it is required for it to return both the identity of the last vertex and a count of how often this particular vertex has been returned. Define a new version of the datatype `G`, containing mutable values, that could be used. [3 marks]

Illustrate the use of this datatype by defining the new versions of `mkgraph` and `last`. [10 marks]

SECTION C

7 Discrete Mathematics

Let us say that a finite partial order $(A, \sqsubseteq)$ is *tree-like* if, for every $a \in A$, the set (of its predecessors) $\{x \in A \mid x \sqsubseteq a \wedge x \neq a\}$ either is empty or has a unique maximal element. Equivalently, pictorially, this holds when the Hasse diagram of A consists of one or more trees.

State which of the following relations on the integers $\{1, 2, \ldots, 10\}$ are tree-like partial orders and give a one-sentence justification.

(*a*) R where $xRy \Leftrightarrow x = y$

(*b*) R where $xRy \Leftrightarrow x \leqslant y$ (here $\leqslant$ is the usual ordering on integers)

(*c*) R where $xRy \Leftrightarrow x$ divides-exactly-into y

(*d*) R where $xRy \Leftrightarrow x = y$ or x is the greatest prime factor of y

[8 marks]

To count the number $C(n)$ of tree-like partial orders of n elements, assume $A = \{1, 2, \ldots, n\}$ and then place each element i in turn into a Hasse diagram starting from 1 and such that no later element $j > i$ is placed such that $j \sqsubseteq i$.

Show that, provided $n > 1$, we have $C(n) = f(n, C(n-1))$ and give the function $f(n, m)$. Provide a base case and thereby solve the recurrence for $C(n)$. [12 marks]

8 Discrete Mathematics

Let A be a set and R a relation on A; also write R^k for the usual k-fold composition of R, i.e. $R^1 = R, R^{k+1} = R \circ R^k$. Let $t(R)$ be the smallest relation which is transitive and has $R \subseteq t(R)$, similarly let $u(R) = \bigcup_{k=1}^{\infty} R^k$.

(a) Show (e.g. by induction on k) that $(\forall k \geqslant 1) R^k \subseteq t(R)$;

(b) deduce $u(R) \subseteq t(R)$.

(c) Show further that $u(R)$ is transitive;

(d) and hence argue that $u(R) = t(R)$.

[8 marks]

Now suppose A has n elements; give a sketch of an algorithm which takes as input an $n \times n$ boolean array V representing R above and produces as output a similar boolean array W representing $t(R)$ above. Give the running time of your algorithm in the form $O(f(n))$. [8 marks]

Find a set A and a relation R on A which is not reflexive, but where $t(R)$ is reflexive. [4 marks]

SECTION D

9 Programming in Modula-3

Describe the facilities in Modula-3 for controlling the repeated execution of a block of code. [10 marks]

Show how `FOR`, `REPEAT` and `WHILE` loops can be emulated by using `LOOP` and `EXIT`. [10 marks]

10 Programming in Modula-3

Describe the facilities in Modula-3 for modelling data by means of `ARRAY`s, `RECORD`s and `REF`erences. [12 marks]

Illustrate your answer by specifying some data types to model a map. This should include points (specifying latitude and longitude), junctions (specifying a point and the roads incident at the junction) and road segments (specifying the junctions at each end, a name and some intermediate points between the ends). [8 marks]

SECTION E

11 Operating Systems

Why does the widespread use of graphical user interfaces (GUIs) make explicit the need for the underlying operating system to support concurrent processes and threads? [2 marks]

Outline the data structures that might be held by an operating system to support the entities that are scheduled (processes or threads). [8 marks]

Describe *one* scheduling algorithm and explain how it would be implemented, based on the data structures you have described above. [7 marks]

What are the implications for scheduling of the need to support new media types such as voice and video? [3 marks]

12 Operating Systems

What information would you expect a filing system to keep on a file? Discuss where this information might be held. [4 marks]

Criticise *each* of the following methods of recording the disc blocks allocated to a file. In each case, indicate the suitability of the method for new media types such as voice and video.

(*a*) chaining in the media

(*b*) detached chain

(*c*) table of pointers

(*d*) extents

[16 marks]