COMPUTER SCIENCE TRIPOS Part IA

Tuesday 3 June 1997 1.30 to 4.30

Paper 2

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

Submit the answers in five **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 **Twenty-part question** (*One mark per part*)

(*a*) Give some ML text to replace `<insert>` in the following:

```
<insert>
fun f g g = g; f x y;
```

to make it into a valid ML program.

(*b*) Given the following ML function definition:

```
fun h x y z = x (y z);
```

what is the type of `h`?

(*c*) Let A be the set $\{1, 2, 3, 4\}$. Exhibit (or state briefly why this is impossible) a relation R on A which is a function whose inverse relation R^{-1} is a partial function, but not a (total) function.

(*d*) Solve the recurrence

$$\begin{aligned} f(0) &= 0 \\ f(1) &= 3 \\ f(n+2) &= 6f(n+1) - 9f(n) \end{aligned}$$

(*e*) It is claimed that there is a bijection from $(A+B) \to C$ to $(A \to C)\times(B \to C)$. Demonstrate this by replacing both "`...`" below with valid ML code which might form such a bijection and its inverse. Just ML is required, not proof.

```
datatype BplusS = left of bool | right of string;
fun LtoR(f: BplusS -> real)
    : (bool->real) * (string->real)
    = ...;
fun RtoL((g: bool->real), (h: string->real))
    : (BplusS -> real)
    = ...;
```

(*f*) Give an equation involving a set named S, such that the smallest (with respect to $\subseteq$) solution for S is the set of values represented by the ML datatype `s`:

```
datatype s = empty | leaf of int | branch of s*s;
```

Your solution may reasonably involve the constructors `empty`, `leaf` and `branch`.

(*g*) Given an integer value between 0 and 9, show an expression in Modula-3 for the equivalent character.

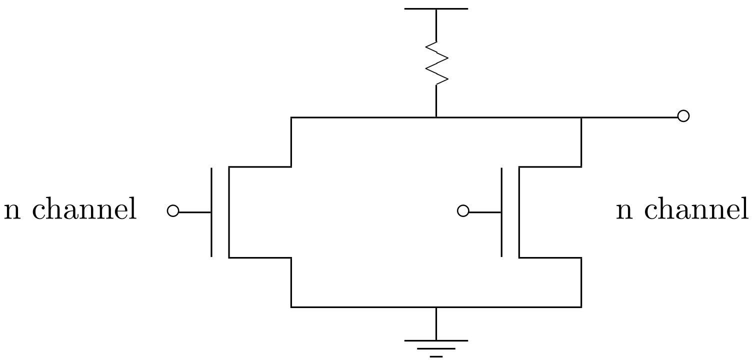
(*h*) What output would the following Modula-3 fragment produce when `P` was called?

```
PROCEDURE P () =
    PROCEDURE Q (VAR a: INTEGER) =
        BEGIN a := 2 * a + b END Q;
    VAR a := 1; b := 2; c := 3;
    BEGIN
        Q (c);
        IO.Put (Fmt.Int (a) & "\n");
        IO.Put (Fmt.Int (b) & "\n");
        IO.Put (Fmt.Int (c) & "\n");
    END P;
```

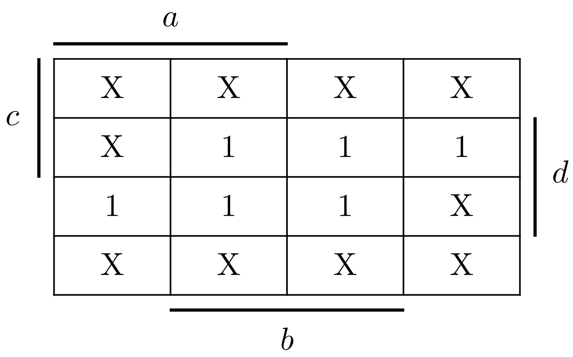
(*i*) What is meant by an *address space*?

(*j*) How are operating system services invoked?

(*k*) What is this?



(*l*) What is the minimum sum of product form for the function specified by the following map in which X indicates *don't care*?



X	X	X	X
X	1	1	1
1	1	1	X
X	X	X	X

(*m*) Identify at least one other factor involved in computer security beyond technical measures such as provision of password access.

(*n*) Name *two* branches of the law under which software ownership can be protected.

(*o*) Provide a generating function $G(\eta)$ that is suitable for a random variable which is distributed Geometric(p).

(*p*) Given that $\mathrm{P}(X{=}0) = \mathrm{P}(Y{=}0) = \frac{1}{4}$ and that $\mathrm{P}(X{=}0 \cap Y{=}0) = \frac{1}{8}$, determine $\mathrm{P}(X{\neq}0 \cap Y{\neq}0)$.

(*q*) Describe the way in which Regular Expressions are constructed.

(*r*) How does the minimum time in which a large software project can be completed depend on the number of staff allocated to it?

(*s*) Give in binary a pair of five-bit numbers which when added give the correct result when considered as unsigned integers but which overflow when considered as two's complement numbers.

(*t*) What is meant by *top–down* programming and how does it differ from *bottom–up* programming?

SECTION B

2 Digital Electronics

What is a *don't care term* and how can such terms arise in practice? [5 marks]

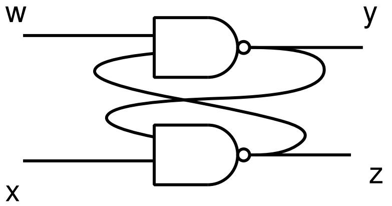
Explain the operation of an edge triggered D-type flip flop, taking care to explain any timing constraints. [5 marks]

Explain the operation of a tristate output. How are such outputs used? [5 marks]

Give the circuit diagram for a CMOS tristate inverter. [5 marks]

3 Digital Electronics

Explain the operation of the following circuit.



[5 marks]

Minimise the following functions f_i. The g_i indicate don't care conditions for the respective f_i; g_i is one for an assignment of values if we do not care what the value of f_i is for that assignment.

(*a*) $f_1 = xyw + \bar{x}\bar{y}zy$
$g_1 = yz\bar{x}w + x\bar{y}wz + \bar{x}\bar{y}\bar{z}w$ [5 marks]

(*b*) $f_2 = xy\bar{z}\bar{w} + x\bar{y}\bar{z}$
$g_2 = xwz + \bar{x}\bar{w}\bar{z}$ [5 marks]

(*c*) $f_3 = wyzv + \bar{x}wzv + xyzw$
$g_3 = x\bar{y}\bar{z}v + xy\bar{z}\bar{v}$ [5 marks]

SECTION C

4 Probability

A gate in a communications network is always in one of two states, *open* or *closed*. At each clock pulse it may change state according to the following rules:

- If it is *open* it remains open with probability $1 - \alpha$ and changes to *closed* with probability α.
- If it is *closed* it remains closed with probability $1 - \beta$ and changes to *open* with probability β.

It may be assumed that $0 < \alpha < 1$ and that $0 < \beta < 1$. Let u_n be the probability that the gate is *closed* just after the nth clock pulse. Derive a difference equation for u_n and justify your derivation. [4 marks]

Let u_0 be the probability that the gate is *closed* initially. Solve your difference equation so as to give u_n in terms of α, β, u_0 and n. [7 marks]

Determine an expression for u_n in the limit as $n \to \infty$ and explain informally why this does not depend on u_0. [3 marks]

Find u_n in each of the four extreme cases: $\alpha = 0$, $\beta = 0$; $\alpha = 0$, $\beta = 1$; $\alpha = 1$, $\beta = 0$; and $\alpha = 1$, $\beta = 1$. Explain the operation of the system in each case. [6 marks]

5 Probability

Suppose $f(x)$ is a probability density function associated with a continuous random variable X and $y(x)$ is a transformation function whose inverse is the function $x(y)$. The derived random variable $Y = y(X)$ will be associated with some probability density function $g(y)$ where:

$$g(y) = f\big(x(y)\big)\left|\frac{dx}{dy}\right|$$

Suppose that X is distributed Uniform(0,1) and, accordingly, has an associated probability density function $f(x)$ given by:

$$f(x) = \begin{cases} 1, & \text{if } 0 \leqslant x < 1 \\ 0, & \text{otherwise} \end{cases}$$

Suppose, further, that Y is required to be distributed as a triangular distribution such that the probability density function $g(y)$ is:

$$g(y) = \begin{cases} 1 + y, & \text{if } -1 \leqslant y < 0 \\ 1 - y, & \text{if } 0 \leqslant y < 1 \\ 0, & \text{otherwise} \end{cases}$$

Determine the transformation function $y(x)$ that satisfies this requirement.

[15 marks]

An experienced gambler notes that when two fair and independent dice are thrown the sum S of the two scores is distributed as a similar, albeit discrete, triangular distribution. S can be transformed into a derived random variable $T = \alpha S + \beta$ (where α and β are constants).

Take the range of values of S as 1 to 13 and observe that the probability of each of these outcomes is zero. What values of α and β ensure that the discrete distribution of T corresponds most closely to the continuous distribution of Y? [5 marks]

SECTION D

6 Computer Perspectives

You start using a new computer, which has an implementation of ML. You write a new ML program which is intended to produce a result of type `int`, i.e. an integer. You run it, and see on your screen (which we assume works properly) the following response:

```
1.99 :  int
```

Discuss the following as possible reasons for the response:

(*a*) your program is faulty

(*b*) the computer's calculation with real numbers is inaccurate

(*c*) the ML implementation is faulty

(*d*) ML is a badly designed language

As part of your answer, discuss the assertion that "a type system is a way of ensuring that every program does what it was supposed to do". [20 marks]

7 Regular Languages and Finite Automata

Janet and John have been asked to produce a formal design for a piece of sequential hardware. Janet starts her design by setting up a regular grammar that characterises the behaviour needed, while John starts with a regular expression. When they have each separately finished that part of their design they decide that they should check to see whether the languages described by their two formalisms are the same.

Explain how (in a systematic way) they can do this. Standard results that they rely on should be stated explicitly and precisely, and comments about the expected costs of performing the comparison may be useful. [20 marks]

8 Software Engineering I

Describe the *waterfall model* of software development and discuss its strengths and weaknesses. [12 marks]

For which of the following projects would it be suitable, and why?

(*a*) an incremental compiler for Java

(*b*) a clinical-record-keeping system for dentists

(*c*) a word-processing package

(*d*) a guidance system for an interplanetary probe

[8 marks]

9 Software Engineering II

You have been invited to start a project to build a program that will check ML programs to see whether they have any missing punctuation marks or other mistakes, so that this check can be performed before the ML code is passed forward for full execution. So far all you know about what is wanted is the above. Without concerning yourself with fine detail of how the checking will be implemented, identify and discuss:

(*a*) questions you might want to ask the client about his needs before deciding on any more details of your design;

(*b*) choices or options available to you when making a detailed project plan;

(*c*) ways of partitioning the whole job into a handful of separate modules that could be implemented more or less independently;

(*d*) plans for testing the code you write and determining whether, when notionally finished, it has met its objectives.

[20 marks]

10 System Architecture

A hard disc and a floppy disc have many common features. Describe the operations which they support in common, as viewed by the filing system software which organises them into files and directories. How are they different? [10 marks]

Describe how a keyboard (or similar device) on a PC or workstation might typically be connected to the central microprocessor and the typical sequence of events when someone presses a key. [10 marks]