COMPUTER SCIENCE TRIPOS Part II (General)
DIPLOMA IN COMPUTER SCIENCE

Tuesday 5 June 2001 1.30 to 4.30

Paper 11 (Paper 2 of Diploma in Computer Science)

Answer **five** *questions.*
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.*

You may not start to read the questions printed on the subsequent pages of this question paper until instructed that you may do so by the Invigilator

1 Digital Electronics

A resource is managed by a system that allocates it to at most one of four requesters for one cycle at a time. The resource is allocated in a "round robin" fashion, so that for example if requester 2 last had the resource, requester 3 has highest priority to acquire it, followed by, in order, requesters 0, 1 and 2. If no one requests the resource it is not allocated, but the system still remembers who last had the resource.

(a) Define input and output variables for the system. [3 marks]

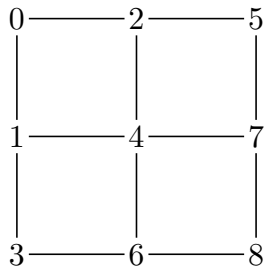
(b) Describe state variables for the system. [3 marks]

(c) Provide equations for the state control and outputs. [10 marks]

(d) Find a minimal sum of products expression for *one* of the outputs. [4 marks]

2 Foundations of Programming

A graph consists of nine numbered vertices and 12 edges thus:



By inspection, it is clear that to get from vertex 0 to vertex 8 there are just six routes in which progress is always from a lower-numbered vertex to one with a higher number.

A programmer wishes to write a Java program to count the routes and begins with the following code:

```
public class Routes
 { private static final int[] first = {1,3,4,6,6,7,8,8};
   private static final int[] second = {2,4,5,0,7,0,0,0};
   private static int[] state = {-1,-1,-1,-1,-1,-1,-1,-1,+1};

   public static void main(String[] args)
    { System.out.println("There are " + tryit(0) + " routes");
    }

   private static int tryit(int vertex)
```

Together the `first` and `second` arrays form a data structure representing the graph. Explain and critically comment on these arrays. [5 marks]

Ultimately, element `v` of the array `state` is intended to show the number of routes from vertex `v` to vertex 8. A value of `-1` indicates that the number is not yet known. The method `tryit(int vertex)` returns the value of `state[vertex]`, calculating and saving this value first if necessary. Supply an appropriate body for this method and explain its operation. [15 marks]

3 Further Java

(*a*) Describe how mutual-exclusion locks provided by the `synchronized` keyword can be used to control access to shared data structures. In particular you should be clear about the behaviour of concurrent invocations of different synchronized methods on the same object, or of the same synchronized method on different objects. [6 marks]

(*b*) Consider the following class definition:

```
class Example implements Runnable {
  public static Object o = new Object();
  int count = 0;

  public void run() {
    while (true) {
      synchronized (o) {
        count ++;
      }
    }
  }
}
```

Show how to start two threads, each executing this `run` method. [2 marks]

(*c*) When this program is executed, only one of the `count` fields is found to increment, even though threads are scheduled preemptively. Why might this be? [2 marks]

(*d*) Define a new class `FairLock`. Each instance should support two methods, `lock` and `unlock`, which acquire and release a mutual exclusion lock such that calls to `unlock` never block the caller, but will allow the longest-waiting blocked thread to acquire the lock. The lock should be *recursive*, meaning that the thread holding the lock may make multiple calls to `lock` without blocking. The lock is released only when a matched number of `unlock` operations have been made.

You may wish to make use of the fact the `Thread.currentThread()` returns the instance of `Thread` that is currently executing. [10 marks]

4 Compiler Construction

Consider the following grammar giving the concrete syntax of a language:

$$
\begin{aligned}
E &\rightarrow id \\
C &\rightarrow E = E; \\
C &\rightarrow \{B\} \\
C &\rightarrow C \ \textbf{repeatwhile} \ E \\
C &\rightarrow \textbf{if} \ E \ \textbf{then} \ C \\
C &\rightarrow \textbf{if} \ E \ \textbf{then} \ C \ \textbf{else} \ C \\
B &\rightarrow B \ C \\
B &\rightarrow C \\
S &\rightarrow C \ eof
\end{aligned}
$$

where C **repeatwhile** E has the same meaning as **do** C **while** E in C or Java.

(a) List the terminals and non-terminals of this grammar and explain the significance of S. [3 marks]

(b) Identify any ambiguities in the above grammar and rewrite it to remove them, ensuring that your new grammar generates exactly the same set of strings. [4 marks]

(c) Specify a suitable abstract syntax, for example by giving a type declaration in a programming language of your choice, which might be used to hold parse trees for this language. [3 marks]

(d) Give *either* a recursive descent parser *or* a characteristic finite state machine (e.g. for SLR(1)) with associated parser for your grammar. Your parser need not return a parse tree—it suffices for your parser either to accept or to reject the input string. [10 marks]

5 Data Structures and Algorithms

(*a*) Outline how you would determine whether the next line segment turns left or right during the Graham scan phase of the standard method of computing the convex hull of a set of points in a plane. [5 marks]

(*b*) Describe in detail an efficient algorithm to determine how often the substring `ABRACADABRA` occurs in a vector of 10^6 characters. Your algorithm should be as efficient as possible. [10 marks]

(*c*) Roughly estimate how many character comparisons would be made when your algorithm for part (*b*) is applied to a vector containing 10^6 characters uniformly distributed from the 26 letters `A` to `Z`. [5 marks]

6 Operating System Foundations

An interprocess communication environment is based on *synchronous* message passing. A server is to be designed to support a moderate number of simultaneous client requests.

Clients send a request message to the server, continue in parallel with server operation, then wait for the server's reply message.

Discuss the design of the server's interaction with the clients. Include any problems you foresee and discuss alternative solutions to them. [20 marks]

7 Mathematics for Computation Theory

(a) Let M be an N-state deterministic finite automaton over the finite alphabet S. Write $l(w)$ for the length of words $w \in S^*$. Suppose that M accepts the word $x \in S^*$, where $l(x) \geqslant N$.

(i) Show that x is a concatenation of words uvw, where $l(u) < N$, $1 \leqslant l(v) \leqslant N$, and M accepts the word $z_k = uv^k w$ for all natural numbers $k \in \mathbb{N}$. [8 marks]

(ii) Hence show that if M accepts an infinite set of words $x \in S^*$, it must accept some word $y \in S^*$ such that $N \leqslant l(y) < 2N$. [4 marks]

(b) A ternary integer representation is defined as follows using a variant of BNF:

```
    <zero> ::= 0
<non-zero> ::= 1 | 2
   <digit> ::= <zero> | <non-zero>
     <nzi> ::= <non-zero> | <nzi> <digit>
 <integer> ::= <zero> | <nzi> | - <nzi>
```

(i) Design a deterministic finite automaton with alphabet $S = \{-, 0, 1, 2\}$ that accepts precisely the valid integer representations. [6 marks]

(ii) Write down a regular expression that denotes the event recognised by your automaton. [2 marks]

[If you wish, you may refer to characters `-`, `0`, `1`, `2` by a, b, c, d respectively.]

8 Computation Theory

(*a*) Define precisely what is meant by the following:

(*i*) $f(x_1, x_2, \ldots x_n)$ is a Primitive Recursive (PR) function of arity n. [5 marks]

(*ii*) $f(x_1, x_2, \ldots x_n)$ is a Total Recursive (TR) function of arity n. [3 marks]

(*b*) Ackermann's function is defined by the following recursive scheme:

$$\begin{aligned} f(0, y) &= S(y) = y + 1 \\ f(x + 1, 0) &= f(x, 1) \\ f(x + 1, y + 1) &= f(x, f(x + 1, y)) \end{aligned}$$

For fixed n define

$$g_n(y) = f(n, y).$$

Show that for all n, $y \in \mathbb{N}$,

$$g_{n+1}(y) = {g_n}^{(y+1)}(1),$$

where $h^{(k)}(z)$ is the result of k repeated applications of the function h to initial argument z. [4 marks]

(*c*) Hence or otherwise show that for all $n \in \mathbb{N}$, $g_n(y)$ is a PR function. [4 marks]

(*d*) Deduce that Ackermann's function $f(x, y)$ is a TR function. [3 marks]

(*e*) Is Ackermann's function PR? [1 mark]

9 Numerical Analysis I

(*a*) What is meant by a *symmetric positive definite matrix*? [3 marks]

(*b*) Verify that $\mathbf{A} = \begin{pmatrix} 2 & 1 \\ 1 & 2 \end{pmatrix}$ is positive definite. [4 marks]

(*c*) The Choleski factorisation $\mathbf{A} = \mathbf{L}\mathbf{D}\mathbf{L}^T$ is to be applied to the solution of $\mathbf{A}\mathbf{x} = \mathbf{b}$, where $\mathbf{b} = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$. It is found that

$$\mathbf{L} = \begin{pmatrix} 1 & \\ \frac{1}{2} & 1 \end{pmatrix}, \quad \mathbf{D} = \begin{pmatrix} 2 & \\ & \frac{3}{2} \end{pmatrix}.$$

The next step in the method is to solve $\mathbf{L}\mathbf{y} = \mathbf{b}$ to get $\mathbf{y} = \begin{pmatrix} 1 \\ \frac{1}{2} \end{pmatrix}$. Form the upper triangular system of equations needed to complete the solution. [4 marks]

(*d*) Solve these equations. [2 marks]

(*e*) What is meant by the *order of convergence* of an iterative process? [1 mark]

(*f*) State the Newton–Raphson formula for solving $f(x) = 0$ for scalar x. What is the order of convergence of this method? [2 marks]

(*g*) This method is used to solve $f(x) = x^2 - 4 = 0$ using IEEE Double Precision with a certain starting value x_0. It is found that the third iterate $x_3 \simeq 2.0006$, and $x_4 \simeq 2.00000009$. Very roughly, how many significant decimal digits of accuracy would you expect in x_5? Explain your answer. [4 marks]

10 Computer Graphics and Image Processing

(*a*) Describe an algorithm to draw a straight line using only integer arithmetic. You may assume that the line is in the first octant, that the line starts and ends at integer co-ordinates, and that the function *setpixel*(x, y) turns on the pixel at location (x, y). [8 marks]

(*b*) Describe Douglas and Pücker's algorithm for removing superfluous points from a line chain. [10 marks]

(*c*) Under what circumstances would it be sensible to employ Douglas and Pücker's algorithm? [2 marks]

11 Natural Language Processing

Define *two* of the following four types of ambiguity, giving examples. For each of these two types, describe *one* possible technique for resolving such ambiguities.

(*a*) Anaphoric coreference ambiguity.

(*b*) Speech act identification ambiguity.

(*c*) Part-of-speech assignment ambiguity.

(*d*) Prepositional phrase attachment ambiguity.

[10 marks each]

12 Software Engineering I

(*a*) State the advantages and disadvantages of the evolutionary model of software development. [8 marks]

(*b*) Is it more, or less, suitable than the waterfall model for safety-critical projects? Justify your answer. [8 marks]

(*c*) If you were managing the evolution of a safety-critical product, what special steps would you take for risk reduction or due diligence reasons? [4 marks]

13 Software Engineering II

(*a*) Describe, with examples, how the choice of programming language, programming tools and libraries can affect the reliability of the software developed using them. [5 marks]

(*b*) Consider the following pair of ML function declarations:

```
fun takew p [] = []
  | takew p (x::xs) = if p x then x :: takew p xs else [];

fun dropw p [] = []
  | dropw p (x::xs) = if p x then dropw p xs else x::xs;
```

Prove `(takew p xs) @ (dropw p xs) = xs` using induction. (Assume that function `p` always terminates.) [8 marks]

(*c*) You have been asked to specify some banking software. A bank account has a *balance* and an overdraft *limit*, subject to the constraints $limit \geqslant 0$ and $balance + limit \geqslant 0$.

(*i*) Write a Z schema to specify the state of a bank account. [2 marks]

(*ii*) Write a Z schema for the operation to withdraw a given positive *amount* from the account. [5 marks]

END OF PAPER