COMPUTER SCIENCE TRIPOS Part IB

Tuesday 6 June 2006 1.30 to 4.30

PAPER 4

*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.*

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

STATIONERY REQUIREMENTS
Script Paper
Blue Coversheets
Tags

1 Economics and Law

You have an opportunity to resell limited supplies of a discontinued model of printer at a good price. You have doubts about how reliable the printers will be. You will sell them through a website, and want to avoid becoming committed to sell more printers than you can acquire. You also want to avoid being liable for defects.

(*a*) How do you plan to set up your website and use the law of contract to achieve these results? [12 marks]

(*b*) How effective will you be? [8 marks]

2 Compiler Construction

Consider the following grammar.

$$E ::= \text{NUM} \mid E \times E \mid E/E \mid E + E \mid E - E \mid (E)$$

(*a*) Show that this grammar is ambiguous. [2 marks]

(*b*) Rewrite the grammar to eliminate ambiguity. [4 marks]

(*c*) Transform the grammar of part (*b*) into an equivalent one that could be used for recursive descent parsing. Explain your answer. [5 marks]

(*d*) Write a recursive descent parser for your grammar. [9 marks]

3 Data Structures and Algorithms

(*a*) Define the minimum spanning tree (MST) of a graph and justify, with counterexamples where appropriate, why the search for it makes sense only on connected, weighted and undirected graphs. [2 marks]

(*b*) Define these MST expressions: *safe edge*, *cut*, *respecting a set of edges*. [3 marks]

(*c*) Describe an efficient MST-finding algorithm, write some clear pseudocode for it and prove its correctness. [9 marks]

(*d*) Say whether *each* of the following two statements is true or false, justifying each answer with a proof or a counterexample.

(*i*) Graph G has a unique MST $\Rightarrow$ For every cut of G, the lightest edge that crosses it is unique.

(*ii*) Graph G has a unique MST $\Leftarrow$ For every cut of G, the lightest edge that crosses it is unique.

[6 marks]

4 Artificial Intelligence I

(*a*) Give a brief description of the way in which *if-then rules* can be used as a basis for knowledge representation and reasoning. What essential elements would you expect to be included in such a system? [3 marks]

(*b*) In the context of such a system, give detailed descriptions, illustrating your answers with specific examples, of the following concepts:

(*i*) pattern matching; [2 marks]

(*ii*) reason maintenance; [2 marks]

(*iii*) forward chaining; [4 marks]

(*iv*) conflict resolution strategies; [4 marks]

(*v*) backward chaining with backtracking. [5 marks]

5 Operating Systems II

In the early 1980s, a new file-system called the *fast file-system* (FFS) was designed for BSD Unix.

(*a*) What *two* problems with the original Unix file system was FFS designed to overcome? [2 marks]

(*b*) For *each* of these problems, describe the solution used, and comment on its effectiveness. [6 marks]

(*c*) From the modern perspective, how appropriate or effective do you believe *each* of these solutions is? Justify your answer. [2 marks]

(*d*) Sketch an efficient design for a file-system to support (soft) real-time access to large multi-media files. Include details of how you would lay out and access files and directories, and how you would perform integrity checking. Be sure to justify any key differences from conventional designs, and to state any assumptions that you make. [10 marks]

6 Mathematical Methods for Computer Science

Consider the N-point Discrete Fourier Transform (DFT) of the sequence $f[n]$ for $n = 0, 1, \ldots, N-1$ given by

$$F[k] = \sum_{n=0}^{N-1} f[n] e^{-2\pi i n k/N}$$

with the inverse DFT given by

$$f[n] = \frac{1}{N} \sum_{k=0}^{N-1} F[k] e^{2\pi i n k/N} \, .$$

(a) Show that $F[k]$ has period N, that is $F[k+N] = F[k]$. [2 marks]

(b) Derive the shift property in the n-domain, namely, that $f[n-m]$ has N-point DFT given by $e^{-2\pi i m k/N} F[k]$. [4 marks]

(c) Define the N-point cyclic convolution, $(f \star g)[n]$, of two sequences $f[n]$ and $g[n]$ by

$$(f \star g)[n] = \sum_{m=0}^{N-1} f[m] g[n-m]$$

and show that $(f \star g)[n]$ has N-point DFT given by $F[k]G[k]$ where $G[k]$ is the N-point DFT of the sequence $g[n]$. [6 marks]

(d) Consider the sequence $f[0] = -1, f[1] = f[2] = f[3] = 1$. Find the 4-point cyclic convolution $f \star f$ by

(i) direct calculation; [4 marks]

(ii) by first writing $f[n]$ in the form $f[n] = c[n] - 2d[n]$ where $c[0] = c[1] = c[2] = c[3] = 1$ and $d[0] = 1, d[1] = d[2] = d[3] = 0$. [4 marks]

You may assume that the binary operation $\star$ distributes over pointwise addition of sequences.

7 Numerical Analysis I

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

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

By drawing a carefully labelled graph, explain the graphical interpretation of this formula. What is the order of convergence? [4 marks]

(b) Consider $f(x) = x^3 + x^2 - 2$. The following table shows successive iterations for each of the three starting values (i) $x = 1.5$, (ii) $x = 0.2$, (iii) $x = -0.5$. Note that, to the accuracy shown, each iteration finds the root at $x = 1$.

n	(i)	(ii)	(iii)
0	1.50000×10^{0}	2.00000×10^{-1}	-5.00000×10^{-1}
1	1.12821×10^{0}	3.95384×10^{0}	-8.00000×10^{0}
2	1.01152×10^{0}	2.57730×10^{0}	-5.44318×10^{0}
3	1.00010×10^{0}	1.70966×10^{0}	-3.72976×10^{0}
4	1.00000×10^{0}	1.22393×10^{0}	-2.56345×10^{0}
5	1.00000×10^{0}	1.03212×10^{0}	-1.72202×10^{0}
6		1.00079×10^{0}	-9.62478×10^{-1}
7		1.00000×10^{0}	1.33836×10^{0}
8		1.00000×10^{0}	1.06651×10^{0}
9			1.00329×10^{0}
10			1.00000×10^{0}
11			1.00000×10^{0}

Sketch the graph of $f(x)$ and show the first iteration for cases (i) and (ii) to show why (i) converges faster than (ii). In a separate sketch, show the first two iterations for case (iii). [Hint: a very rough sketch will suffice for case (iii).] [10 marks]

(c) Now consider $f(x) = x^4 - 3x^2 - 2$. Calculate two Newton–Raphson iterations from the starting value $x = 1$. Comment on the prospects for convergence in this case. [6 marks]

8 Concurrent Systems and Applications

(*a*) To which components of a class definition is it legal to apply the `strictfp` modifier in Java and what effect does it have in each case? [3 marks]

(*b*) When the `transient` modifier is applied to a field of type *object reference to an array of Objects* in a class definition, Java's default Serialization mechanism will omit the array itself and all of the elements in the array from a serialized representation. A useful and reusable class would offer a *partially transient array*: a class containing an ordered list of elements indexed by integers from zero upwards, which allows the programmer to toggle each item separately between being transient and non-transient. The class should provide methods to get and set the values in each position of the array and to indicate that an element should behave as though it is/is not transient.

(*i*) Define a *generic* Java interface, `PartiallyTransientArray`, appropriate for this purpose. [4 marks]

(*ii*) Give the definition of a *generic* class providing the functionality of a partially transient array ensuring that, when serialized, the class and (only) the non-transient elements are included in the output. It is sufficient to state the size of the array at construction-time and for it to be unchangeable thereafter. [6 marks]

(*c*) Explain, drawing examples from your partially transient array class, how formal type parameters can be restricted in the Java language using the `extends` and `super` keywords. [4 marks]

(*d*) What are the drawbacks of the "Generics" features of the Java language for providing type-polymorphism? [3 marks]

9 Computation Theory

(*a*) Define the collection of *primitive recursive* functions. [6 marks]

(*b*) Why is a primitive recursive function always total? [1 mark]

(*c*) Show that the function m from $\mathbb{N}^2$ to $\mathbb{N}$ given by

$$m(x,y) = \begin{cases} x - y & \text{if } x \geq y \\ 0 & \text{if } x < y \end{cases}$$

is primitive recursive. [3 marks]

(*d*) Define the collection of *partial recursive* functions. [3 marks]

(*e*) What is meant by a *total recursive* function? [1 mark]

(*f*) Show that there exist total recursive functions that are not primitive recursive. Any standard results about register machines or recursive functions that you use need not be proved, but should be clearly stated. [6 marks]

10 Introduction to Security

(*a*) Alice and Bob participate in a public-key infrastructure that enables them to exchange legally binding digital signatures.

(*i*) Name *two* reasons why, for some purposes, Alice might prefer to use a message authentication code, instead of a digital signature, to protect the integrity and authenticity of her messages to Bob. [4 marks]

(*ii*) Outline a protocol for protecting the integrity and authenticity of Alice's messages to Bob that combines the benefits of a public-key infrastructure with those of using a message authentication code. [4 marks]

(*b*) Your colleague proposes a new way for constructing a message authentication code using a block cipher $E : \{0,1\}^{64} \times \{0,1\}^{128} \to \{0,1\}^{128}$. He takes the n-bit input message M, appends $p = 64 \cdot \lceil n/64 \rceil - n$ zero-bits, and splits the result into $k = (n+p)/64$ 64-bit blocks $M_1||M_2||\ldots||M_k = M||0^p$. He then calculates the message authentication code as

$$C_K(M) = E_{M_1}(E_{M_2}(E_{M_3}(\ldots E_{M_k}(K)\ldots)))$$

where K is the 128-bit secret key shared between sender and recipient. Show *two* different ways in which an attacker who observes a pair $(M, C_K(M))$ can, without knowing K, create a new pair $(M', C_K(M'))$ with $M' \neq M$. [6 marks]

(*c*) Show how a 128-bit message authentication code $C_K(M)$ with 64-bit key K can be constructed for an n-bit long message M using

(*i*) a secure hash function $H : \{0,1\}^* \to \{0,1\}^{256}$, such as SHA-256; [2 marks]

(*ii*) a block cipher $E : \{0,1\}^{128} \times \{0,1\}^{256} \to \{0,1\}^{256}$. [4 marks]

END OF PAPER