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

Wednesday 5 June 2002 1.30 to 4.30

Paper 12 (Paper 3 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.*

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 Data Structures and Algorithms

You have available a 20 Gbyte disc on which you need to hold an indexed sequential file consisting of variable length records each having a 20 byte key. Records, including the key, are typically 500 bytes long but never exceed 1000 bytes. The total size of all the records is never more than 10 Gbytes.

(*a*) Suggest, in detail, how you would represent this file on the disc. You should choose an organisation that allows

(*i*) efficient insertion of new records,

(*ii*) efficient updating of existing records identified by key, and

(*iii*) efficient inspection of all records in key order.

[14 marks]

(*b*) If the total size of the database is 10 Gbytes, estimate, for your organisation of the file, how many disc transfers would be needed to access a record with a given key, and estimate how many transfers would be required to read the entire database in sequential order. [6 marks]

2 Computer Design

Modern dynamic random access memories (e.g. DRAM, SDRAM and RAMBUS) all support burst mode read and write access.

(*a*) Give an outline of the bus activity for a burst mode read access. [4 marks]

(*b*) Explain the difference between a direct mapped cache and an associative cache. [4 marks]

(*c*) What cache line replacement policies are typically used for a direct mapped cache and a set associative cache? [4 marks]

(*d*) Why are caches able to exploit burst mode accesses, and why is a write buffer often used? [4 marks]

(*e*) What is *bus snooping* and what does it achieve? [4 marks]

3 Digital Communication I

Consider the real-time transport of audio across a network.

(*a*) What are the advantages of digitising the audio? [5 marks]

(*b*) What are the disadvantages and how can they be mitigated? [5 marks]

(*c*) What characteristics of the end-to-end channel across the network would be desirable, and how are these different from those which would be desirable for time-insensitive data? [5 marks]

(*d*) Discuss the applicability of asynchronous and synchronous multiplexing in carrying real-time audio traffic. [5 marks]

4 Business Studies

(*a*) Distinguish between top-down, bottom-up and spiral (rapid prototype) development methodologies. Illustrate your answer with reference to an example of designing a building. [5 marks]

(*b*) You are in charge of commissioning the design of a new building, such as the new Computer Laboratory building. Draw up a high-level GANTT chart for this task up to the letting of the building contract. [10 marks]

(*c*) Discuss what monitoring and quality control procedures might apply to the design process. How will you get the agreement of the various stakeholders? [5 marks]

5 Comparative Programming Languages

This question concerns the representation of parse tree nodes for expressions composed of integer constants, identifiers, and integer operators for addition, subtraction, multiplication and division. In a typeless language, such as BCPL, each node can be implemented as a vector whose first element holds an integer identifying the node operator. The size of the vector and the kinds of value held in the other elements then depends on this node operator.

(*a*) Complete the description of how you would represent such integer expressions in a typeless language. [5 marks]

(*b*) Suggest how you would represent such integer expressions in C and *either* ML *or* Java. [10 marks]

(*c*) Briefly discuss the relative merits of your C data structure compared with that used in the typeless approach. [5 marks]

6 Compiler Construction

(*a*) Assuming a Java type is given to each variable, state a method by which an overloaded operator (such as `+`,`-` etc.) in a Java program can be determined to be an `int`, `real` or other operator. [3 marks]

(*b*) Explain, using pseudo-code as appropriate, how to convert a syntax-tree into stack code such as that used in the JVM. For the purposes of this question, you only need consider trees representing bodies of `void`-returning functions, and these bodies only as consisting of a list of statements of the form `int` x `=` e`;` or $x = e$; where x ranges over variables and e over expressions; expressions contain variables, integer constants, the binary operator `+` and static method invocations. [10 marks]

(*c*) Show how a sequence of simple stack code instructions, such as those used in your answer to part (*b*) above, can be translated into a sequence of instructions for a register-oriented architecture of your choice, for example ARM or Pentium. [7 marks]

7 Prolog for Artificial Intelligence

(*a*) Give a simple definition of the Prolog predicate `dfx` that can perform symbolic differentiation with respect to the variable `x` of expressions composed of integers (e.g. `0`, `1`, ...), symbolic constants (e.g. `a`, `b`, ...), symbolic variables (e.g. `x`, `y`, ...) and the operators `+`, `-` and `*`, for addition, subtraction and multiplication. The first argument of `dfx` is the expression to differentiate and the second argument is the result. Your definition need not perform any simplification of the result. [6 marks]

(*b*) Trace the execution of the call: `dfx(x*x-2, R)`. [2 marks]

(*c*) Now modify your definition so that it simplifies the result by the applications of rewriting rules such as: `1*x`⇒`x` and `x-0`⇒`x`. [8 marks]

(*d*) Discuss to what extent, if any, either of your predicates could be used to integrate an expression. [4 marks]

8 Databases

(*a*) Describe the relational model of data. [6 marks]

(*b*) Explain the following concepts in relational databases:

(*i*) entity integrity constraint; [1 mark]

(*ii*) foreign key and how it can specify a referential integrity constraint between two relations; [4 marks]

(*iii*) semantic integrity constraint. [1 mark]

(*c*) (*i*) What is a functional dependency? [1 mark]

(*ii*) Define Boyce-Codd Normal Form (BCNF). [3 marks]

(*iii*) Define Third Normal Form (3NF). [3 marks]

(*iv*) What is the relationship between BCNF and 3NF? [1 mark]

9 Numerical Analysis II

(*a*) In the Chebyshev characterisation theorem, the best L_∞ approximation to $f(x)$ over a closed finite interval by a polynomial $p_{n-1}(x)$ of degree $n-1$ has the property that the maximum error $|e(x)|$ is attained at M distinct points ξ_j such that $e(\xi_j) = -e(\xi_{j-1})$. What is M? [2 marks]

(*b*) Let $x = m \times 2^k$ represent a normalised number in a floating-point implementation. When computing $\sqrt{x}$ show how the domain of the problem can be reduced to $x \in [1, 4)$. Find the coefficients a, b which minimise $||e(x)||_\infty$ over $[1, 4]$ where $e(x) = ax + b - \sqrt{x}$. [8 marks]

(*c*) Taking full account of symmetry, describe the form of the best polynomial approximation $p(x)$ to x^4 over $[-1, 1]$ by a polynomial of lower degree. Sketch x^4 and $p(x)$, showing the extreme values of $|e(x)|$ where $e(x) = x^4 - p(x)$. Hence compute the coefficients of $p(x)$. [10 marks]

10 Introduction to Functional Programming

(*a*) Give a recursive definition of an ML datatype `'a tree` of binary trees consisting of nodes where data items are stored. Each such node is either a leaf or a branch node with left and right trees as branches. [3 marks]

(*b*) Write a recursive function `size` of type `'a tree -> int` that returns the number of nodes of a given tree. [4 marks]

(*c*) Write an iterative function `isize` of type `int * 'a tree -> int` which satisfies the following identity for all integers n and all trees t

$$\texttt{isize}(n, t) = \texttt{size}(t) + n \tag{1}$$

[6 marks]

(*d*) Prove, by structural induction, that the identity (1) holds for the two functions you defined. [7 marks]

11 Computer Vision

The following very useful operator is often applied to an image $I(x, y)$ in computer vision algorithms, to generate a related "image" $g(x, y)$:

$$g(x,y) = \int_\alpha \int_\beta \nabla^2 e^{-\left((x-\alpha)^2+(y-\beta)^2\right)/\sigma^2} I(\alpha,\beta)\ d\alpha\ d\beta$$

where

$$\nabla^2 = \left(\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}\right)$$

(a) Give the general name for this type of mathematical operator, and the chief purpose that it serves in computer vision. [2 marks]

(b) What image properties should correspond to the zeroes of the equation, i.e. those points (x, y) in the image $I(x, y)$ where the above result $g(x, y) = 0$? [3 marks]

(c) What is the significance of the parameter σ? If you increased its value, would there be more or fewer points (x, y) at which $g(x, y) = 0$? [3 marks]

(d) Describe the effect of the above operator in terms of the two-dimensional Fourier domain. What is the Fourier terminology for this image-domain operator? What are its general effects as a function of frequency, and as a function of orientation? [4 marks]

(e) If the computation of $g(x, y)$ above were to be implemented entirely by Fourier methods, would the complexity of this computation be greater or less than the image-domain operation expressed above, and why? What would be the trade-offs involved? [4 marks]

(f) If the image $I(x, y)$ has 2D Fourier Transform $F(u, v)$, provide an expression for $G(u, v)$, the 2D Fourier Transform of the desired result $g(x, y)$ in terms of only the Fourier plane variables $u, v, F(u, v)$, and the above parameter σ. [4 marks]

12 Complexity Theory

(a) Give a precise definition of what is meant by the statement that a decision problem A is *polynomial-time reducible* to a decision problem B. [2 marks]

(b) Consider the following three decision problems on graphs.

Connect—the collection of graphs G such that there is a path between any two vertices of G.

Hamilton—the collection of graphs that contain a Hamiltonian cycle.

non-3-colour—the collection of graphs that are not 3-colourable.

For each of the following statements, state whether it is true, false or an unresolved open question. Give a brief justification for your answer.

(i) **Connect** is decidable by a polynomial time algorithm.

(ii) **Hamilton** is decidable by a polynomial time algorithm.

(iii) **non-3-colour** is decidable by a polynomial time algorithm.

(iv) **Connect** is polynomial-time reducible to **Hamilton**.

(v) **Hamilton** is polynomial-time reducible to **non-3-colour**.

(vi) **non-3-colour** is polynomial-time reducible to **Connect**.

[3 marks each]

END OF PAPER