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

Thursday 9 June 2005 1.30 to 4.30

Paper 13 (Paper 4 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

STATIONERY REQUIREMENTS
Script Paper
Blue Coversheets
Tags

1 Data Structures and Algorithms

A certain program has to maintain an array, `count`, of N counters which are all initialised to zero. The value of counter i can be incremented by one by the call: `increment(i)`, and this is the only way the program changes counter values. Two variables, `mincount` and `maxcount`, must always hold the smallest and largest of the counter values whenever the point of execution is not within the function `increment`. You may assume that `increment` is called about $1000N$ times when the program is run and that its argument is typically uniformly randomly distributed between 1 and N, but on some runs it cycles through the numbers 1 to N in order 1000 times.

(*a*) Describe, in detail, an efficient data structure and algorithm to use when N is expected to be about 10. [5 marks]

(*b*) Describe, in detail, an alternative data structure and algorithm to use when N is about a million. [10 marks]

(*c*) Suppose your algorithm for (*a*) above were used when $N = 10^6$, estimate how much slower it would be compared with your algorithm for (*b*). [5 marks]

2 Computer Design

(*a*) What is a *data cache* and why is it vital for high performance processors? [5 marks]

(*b*) What is a *cache line* and how big is it likely to be? [3 marks]

(*c*) What is a *page* and how big is it likely to be? [3 marks]

(*d*) What is a *snoopy cache* and in what system is it likely to be used? [3 marks]

(*e*) A computer system has the following memory parameters:

memory level	access time in clock cycles
first level cache	1
second level cache	3
main memory	100

If the probability of a hit in the first level cache is 80%, what hit rate do we need in the second level cache if the mean access time is to be two clock cycles? For this part, use the approximation that the cache line length is just one machine word. [6 marks]

3 Digital Communication I

(*a*) Describe *on–off* flow control. In what circumstances is it appropriate? [4 marks]

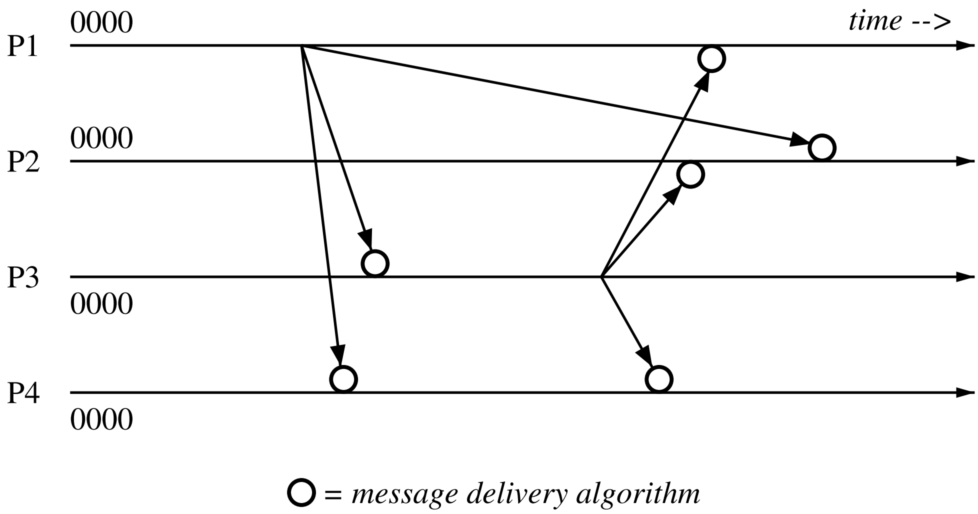
(*b*) Describe the operation of *window-based* flow control. [4 marks]

(*c*) What happens if window-based flow control is used on a flow passing through a highly loaded resource (e.g. router) that is not participating in the flow control protocol? [4 marks]

(*d*) How is this addressed in the Internet? [4 marks]

(*e*) What are the advantages and disadvantages of having Internet routers participate in window-based flow control of every TCP connection? [4 marks]

4 Distributed Systems



The above diagram represents a process group that communicates by means of multicast messages. At each process-hosting node, message delivery software decides whether a given incoming message should be delivered to the process or buffered for later delivery. This is achieved by the use of vector clocks.

(*a*) Describe, by means of the above example, the vector clock algorithm for delivery of messages in causal order. [15 marks]

(*b*) By means of a similar example, show that total ordering of messages is not achieved by this algorithm. [5 marks]

5 Compiler Construction

(*a*) Give a program which gives different results according to whether dynamic or static binding is used. [3 marks]

(*b*) Summarise two different methods of implementing functions which have free variables statically bound in an outer nested function, paying particular attention to (*i*) the lifetime of any storage used and (*ii*) any language restrictions which the implementation requires. Note also, for each case, what the effect is of assignment to such free variables, e.g. when calls to `g()` and `h()` are interleaved in

```
f(int x)
{   int a = x+7;
    int g() { a++; }
    int h(int y) { return a+y; }
    ...
}
```

[8 marks]

(*c*) Give a program in which a function is called recursively without any value or function being defined recursively. [3 marks]

(*d*) Give a program which produces three distinct results according to whether parameters are passed by value, by reference, or by value-result. [3 marks]

(*e*) Explain the likely storage layouts for Java objects *p* and *q* respectively of classes `P` and `Q` defined by

```
class P           { int a,b; int f() { <body1> } }
class Q extends P { int c; int f() { <body2> }
                           int g() { <body3> } }
```

[3 marks]

6 Computer Graphics and Image Processing

(*a*) In ray tracing, once we have determined where a ray strikes an object, the illumination at the intersection point can be calculated using the formula:

$$I = I_a k_a + \sum_i I_i k_d (\mathbf{L}_i \cdot \mathbf{N}) + \sum_i I_i k_s (\mathbf{R}_i \cdot \mathbf{V})^n.$$

Explain what real effect each of the three terms is trying to model, how accurately it models the real effect, and explain what each of the following symbols means, within the context of this formula:

$$I,\ I_a,\ i,\ I_i,\ k_a,\ k_d,\ k_s, \mathbf{L}_i,\ \mathbf{N},\ \mathbf{R}_i,\ \mathbf{V},\ n.$$

[12 marks]

(*b*) Compare and contrast the ray tracing and z-buffer algorithms. [8 marks]

7 Comparative Programming Languages

(*a*) Many computer scientists believe that languages with strong compile-time type checking are better than those that are typeless, dynamically typed, or are weakly type checked. Discuss the reasons for this view. [7 marks]

(*b*) If strictly-checked data types are seen as good, discuss whether augmenting a language with many more primitive data types is better. Consider, in particular, the possibility of incorporating into a language such as Java many new numerical types such as packed decimal of various precisions, scaled arithmetic, and new types to hold values representing distance, mass and time. How would these additions affect the readability and reliability of programs? [7 marks]

(*c*) Some languages allow different modes of calling of function arguments, such as call by value, call by reference and call by name. Discuss the advantages and disadvantages of incorporating the argument calling modes into the data types of functions. [6 marks]

8 Databases

(*a*) OLAP and OLTP.

(*i*) What is *on-line transaction processing* (OLTP)? [2 marks]

(*ii*) What is *on-line analytic processing* (OLAP)? [2 marks]

(*iii*) If you were designing a relational database system, how would your approach to schema design differ for OLTP and OLAP systems? [3 marks]

(*iv*) In OLAP, what is the meaning of the terms *drill down*, *roll up*, and *slice*? [3 marks]

(*v*) What is a *star schema*? [1 mark]

(*b*) Suppose we have the following relational schema,

Supplier(<u>sid:integer</u>, name:string, postcode:string)
Parts(<u>pid:integer</u>, name:string, description:string)
SuppliedBy(<u>sid:integer, pid:integer</u>, weight:integer)

where the underlined attributes represent the primary keys of the associated relation. The table *SuppliedBy* implements a relationship between suppliers and parts — indicating which parts are supplied by which supplier — using foreign keys pointing into the *Parts* and *Supplier* tables. The *weight* attribute is the parts weight in grams.

Write an SQL query that will return a list, without duplicates, of all postcodes associated with suppliers of parts less than one kilogram in weight. [5 marks]

(*c*) Define and explain the ACID properties of database transactions. [4 marks]

9 Numerical Analysis II

(a) Explain the term *positive semi-definite*. If $\mathbf{A}$ is a real square matrix show that $\mathbf{A}^T\mathbf{A}$ is symmetric and positive semi-definite. [3 marks]

(b) How is the l_2 *norm* of $\mathbf{A}$ defined? State Schwarz's inequality for the product $\mathbf{Ax}$. [2 marks]

(c) Describe briefly the properties of the matrices $\mathbf{U}$, $\mathbf{W}$, $\mathbf{V}$ in the *singular value decomposition* $\mathbf{A} = \mathbf{UWV}^T$. [3 marks]

(d) Let $\hat{\mathbf{x}}$ be an approximate solution of $\mathbf{Ax} = \mathbf{b}$, and write $\mathbf{r} = \mathbf{b} - \mathbf{A}\hat{\mathbf{x}}$, $\mathbf{e} = \mathbf{x} - \hat{\mathbf{x}}$. Derive a computable estimate of the relative error $\|\mathbf{e}\|/\|\mathbf{x}\|$ in the approximate solution, and show how this may be used with the l_2 norm. [8 marks]

(e) Suppose $\mathbf{A}$ is a 7×7 matrix whose singular values are 10^2, 10^{-4}, 10^{-10}, 10^{-16}, 10^{-22}, 10^{-29}, 10^{-56}. Construct the matrix $\mathbf{W}^+$ that you would use (i) if *machine epsilon* $\simeq 10^{-15}$, and (ii) if *machine epsilon* $\simeq 10^{-30}$. [4 marks]

10 Introduction to Functional Programming

(*a*) Define a function `subsets` in ML with the type

```
subsets :  'a list -> 'a list list
```

which treats its input list as a set of n elements and returns a list of all 2^n subsets of that set. [5 marks]

(*b*) Define an exception `NoFit` with no arguments, and an exception `Success` whose constructor takes an `int list` as an argument. [1 mark each]

(*c*) Use `subsets` to define a function `knapsack` with the type

```
knapsack :  int -> int list -> int list
```

which finds a subset of the `int`s in its second argument whose sum is exactly the first argument. Use at least one functional (`map`, `foldl`, etc.) in the definition of `knapsack`. The `NoFit` exception should be raised in the event that no solution exists. [5 marks]

(*d*) Define another version of `knapsack` called `knapsack2` with the type

```
knapsack2 :  int -> int list -> unit
```

`knapsack2` should use simple recursion to generate its candidate solutions on-the-fly as it tests them, and should raise `Success` with the solution on success or return `unit` if no solution exists. [8 marks]

11 Natural Language Processing

(*a*) The Figure below shows feature structures corresponding to lexical entries for *snores* and *he*.

$$
\left[\begin{array}{ll}
\textsc{head} & \left[\ \textsc{cat}\ \textbf{verb}\ \right] \\
\textsc{comp} & \textbf{filled} \\
\textsc{spr} & \left[\begin{array}{ll} \textsc{head} & \left[\ \textsc{cat}\ \textbf{noun}\ \right] \\ \textsc{sem} & \left[\ \textsc{index}\ \boxed{1}\ \right] \end{array}\right] \\
\textsc{sem} & \left[\begin{array}{ll} \textsc{pred} & \textbf{snore\_v} \\ \textsc{arg1} & \boxed{1} \end{array}\right]
\end{array}\right]
\qquad
\left[\begin{array}{ll}
\textsc{head} & \left[\ \textsc{cat}\ \textbf{noun}\ \right] \\
\textsc{comp} & \textbf{filled} \\
\textsc{spr} & \textbf{filled} \\
\textsc{sem} & \left[\begin{array}{ll} \textsc{index} & \boxed{1} \\ \textsc{pred} & \textbf{pron} \\ \textsc{arg1} & \boxed{1} \end{array}\right]
\end{array}\right]
$$

These structures can be combined using a grammar rule to give a feature structure corresponding to the phrase *he snores* with a semantic structure equivalent to pron(x) $\wedge$ snore_v(x). Give this grammar rule as a feature structure and show the results of applying the rule to the structures in the Figure. [7 marks]

(*b*) Syntactically the verb *rains* takes the pleonastic pronoun *it* as subject but semantically it has no arguments. Give possible feature structures for *rains* and pleonastic *it*. Show how ungrammatical sentences such as *he rains* are avoided, mentioning any modifications to the lexical entries in the Figure that might be necessary. [6 marks]

(*c*) Selectional restrictions can be used to block parses of semantically anomalous sentences such as:

The pebble snores.
The pebble wrote a book.
The dog wrote a book.

Describe how selectional restrictions might be encoded in a feature structure grammar. [7 marks]

12 Complexity Theory

(*a*) When defining space-complexity we use two-tape Turing Machines, with a read-only tape for input data and a read–write working tape. We count only the space used on the work tape. What difference would be made if we worked with standard one-tape Turing machines, loaded the input data onto the tape to start with and measured space in terms of total tape cells touched by the end of the computation? [4 marks]

(*b*) Comment on the following: "The problem of finding a factor of a number N is NP, because if we have a factor P of N we can do a simple trial division and check it in time related to $\log(N)$. Thus finding factors of numbers of the form $2^p - 1$ (these are known as Mersenne Numbers) is a problem in the class NP". [2 marks]

(*c*) Define the class co-NP. State an example of a problem that lies in it. [2 marks]

(*d*) What is a *witness function* for an NP problem? Why might some problem such as 3-SAT have many different witness functions associated with it? [2 marks]

(*e*) Give and justify a relation between $NTIME(f(n))$ and $SPACE(f(n))$. [4 marks]

(*f*) Matchings on bi-partite graphs can be found in polynomial time. The matching problem on tri-partite graphs is known to be NP-complete. Does this suggest that the corresponding problem with a graph whose nodes are partitioned into four sets ("quad-partite" matching) is liable to be exponential in complexity? Justify your answer. [4 marks]

(*g*) Comment on the following proposition: "Determining which player can force a win from a given starting position in the game of Chess is an NP problem because given any sequence of moves it will be easy to verify that they are all legal moves and easy to see who wins at the end of them." [2 marks]

END OF PAPER