COMPUTER SCIENCE TRIPOS Part IA

Monday 1 June 1998 1.30 to 4.30

Paper 1

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

Submit the answers in six **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 Foundations of Computer Science

Code the function `least(n,xs)`, which returns the least n elements of the list xs of real numbers. The result does not have to be sorted. You may assume $n \leqslant \texttt{length}(xs)$.

[Hint: Function `least` is a simple modification of quicksort. A solution that works by sorting the entire list will receive little credit. You may take the function `length` as given.]

[10 marks]

2 Discrete Mathematics

What is a relation on a set? [1 mark]

What is an equivalence relation on a set? [3 marks]

Prove that an equivalence relation partitions a set into disjoint equivalence classes. [4 marks]

Given $n \in \mathbb{Z}$ define a relation R on $\mathbb{N}$ by $aRb \iff b - a = n$. For what values of n is R an equivalence relation? What are the equivalence classes? [2 marks]

3 Programming in Java

Give a brief explanation of each of the following aspects of Java:

(*a*) the difference between `>>` and `>>>`;

(*b*) the possibility that in some program the test `(a == a)` might return the value `false` for some variable `a`;

(*c*) the keywords `final` and `finally`;

(*d*) the expression `"three" + 3` and other expressions of a generally similar nature;

(*e*) the meaning of, or errors in (whichever case is relevant!)

```
...
int [10] a;
for (int i=1; i<=10; ++i)
    a[i] = 1-a[i];
...
```

[2 marks each]

4 Operating Systems

Typical memory architectures combine both paging and segmentation for the management of virtual memory. Contrast the two approaches with regard to their support for dynamic memory allocation, efficiency of memory use, support for code/data sharing, and data protection. [10 marks]

SECTION B

5 Foundations of Computer Science

Discuss how lazy lists and mutable lists can be coded in ML. How do they compare with ML's built-in lists? Illustrate your answer by considering the operations of reversing a list and of concatenating two lists. Your discussion should mention the main programming hazards. [6 marks]

The function `odds` is to return the list of alternate elements of its input. For example, `odds`$[a, b, c, d, e] = [a, c, e]$ and `odds`$[a, b] = [a]$. Code `odds` using

(*a*) ordinary ML lists [3 marks]

(*b*) lazy lists [5 marks]

(*c*) mutable lists (as an imperative operation — so that `odds` has type `'a mlist -> unit` for a suitable datatype `'a mlist` of mutable lists, to be defined) [6 marks]

6 Foundations of Computer Science

What does $O(g(n))$ mean, and what is its relevance to programming? (Describe both advantages and limitations.) [5 marks]

Consider the following ML declarations, for tree-like expressions:

```
datatype 'a expr = Join of 'a expr * 'a expr
                 | Tip  of 'a;

fun flatten (Tip x)        = [x]
  | flatten (Join (e1,e2)) = flatten e1 @ flatten e2;
```

The *size* of an expression is the number of `Tip`s it contains. State the complexity of `flatten`(e), measured in cons operations, as a function of the size of e:

(*a*) in the worst case [3 marks]

(*b*) in the average case [4 marks]

(*c*) in the best case [3 marks]

Code a function `flat` such that `flat`(e) = `flatten`(e) for all e, justifying this claim. Show that `flat`'s worst-case complexity is linear. [5 marks]

SECTION C

7 Discrete Mathematics

State and prove the Chinese Remainder Theorem concerning the simultaneous solution of a pair of congruences to co-prime moduli and the uniqueness of that solution. [10 marks]

An early form of public key encryption worked as follows. A person, R, wishing to receive secret messages, selected two large primes, p and q also co-prime to $p-1$ and $q-1$, and published their product, $n = p \times q$. Another person, S, wishing to send a message m to R, encoded it as $s = m^n (\mathrm{mod}\ n)$.

Show how to calculate inverses a and b so that $ap \equiv 1(\mathrm{mod}\ q-1)$ and $bq \equiv 1(\mathrm{mod}\ p-1)$. By considering $s^a(\mathrm{mod}\ q)$ and $s^b(\mathrm{mod}\ p)$ and recalling the Fermat–Euler theorem, show how R could recover the original message, m. State clearly any other results that you use. [10 marks]

8 Discrete Mathematics

What is a partial order on a set? What is a total order?

Given two totally ordered sets $(A, \leqslant_A)$ and $(B, \leqslant_B)$, define the Product Order and Lexicographic Order on $A \times B$ and show that they are partial orders. Show that the Lexicographic Order is a total order and that it contains the Product Order. [10 marks]

Let S be the set of functions from $\mathbb{N}$ to $\{0, 1\}$. Define a relation F on S by

$$(f, g) \in F \iff \forall n \in \mathbb{N}.\ f(n) \leqslant g(n).$$

Show that F is a partial order.

Define a relation G on S by

$$(f, g) \in G \iff \sum_{n \in \mathbb{N}} f(n) 10^{-n} \leqslant \sum_{n \in \mathbb{N}} g(n) 10^{-n}.$$

Show that G is a total order and that it contains F. [10 marks]

SECTION D

9 Programming in Java

A complete Java program may use the same name for several different methods or variables. Java has a number of features that allow the user to prevent such re-use of names from causing chaos. Describe these under the headings:

(*a*) scope rules within individual functions; [6 marks]

(*b*) visibility of method names within classes, and the effects of inheritance; [8 marks]

(*c*) avoiding ambiguity when referring to the names of classes. [6 marks]

10 Programming in Java

Write fragments of Java definitions, declarations or code to achieve each of the following effects. You are not expected to show the whole text of a complete program — just the parts directly important for the task described — and you may describe in words rather than Java syntax any supporting definitions or context that you will want to rely on. Clarity of explanation will be viewed as at least as important as syntactic accuracy in the marking scheme. It is also understood that names of methods from the standard Java class libraries are things that programmers check in on-line documentation while writing code, so if you need to use any of these you do not need to get their names or exact argument-format correct provided that (*a*) you describe clearly what you are doing and (*b*) your use is correct at an overview level.

(*a*) Take a `long` argument called `x` and compute the `long` value obtained by writing the 64 bits of `x` in the opposite order. [6 marks]

(*b*) Define a class that would be capable of representing simple linked lists, where each list-node contains a string. You should show how to traverse such lists, build them and how to reverse a list. In the case of the list reversing code please provide two versions, one of which creates the reversed list by changing pointers in the input list, and another which leaves the original list undamaged and allocates fresh space for the reversed version. [8 marks]

(*c*) Cause a line to appear in the window of an applet running from the bottom left of the window towards the top right. Your line should remain visible if the user obscures and then re-displays the window, but you can assume that the size of the windows concerned will be fixed at 100 by 100 units. [6 marks]

SECTION E

11 Operating Systems

As well as storage and retrieval of data, the functions of a filing system include:

- naming and name resolution
- access control
- existence control
- concurrency control

For *each* of the above

(*a*) Briefly define the function. [1 mark each]

(*b*) Discuss how, and in which filing system component, the function may be provided. Use examples from one or more real operating systems to illustrate your answer. [4 marks each]

12 Operating Systems

What is meant by the term *demand paging* in a virtual memory management system, and how is it implemented? [5 marks]

List *five* techniques which the operating system can use to enhance the efficiency of demand paging. [5 marks]

Suppose the page table for the currently executing process is as follows. All values are decimal, and indexes are numbered from zero. Addresses are memory byte addresses. The page offset is 10 bits.

Virtual Page #	Valid bit	Reference bit	Modify bit	Page Frame #
0	1	1	0	4
1	1	1	1	7
2	0	0	0	–
3	1	0	0	2
4	0	0	0	–
5	1	0	1	0

Describe exactly how, in general, a virtual address generated by the CPU is translated into a physical address, with the aid of a diagram. [4 marks]

To what physical address (if any) would the following virtual addresses correspond?

(*a*) 1052

(*b*) 2221

(*c*) 5499

[6 marks]