COMPUTER SCIENCE TRIPOS Part Iʙ

Monday 5 June 2000 1.30 to 4.30

Paper 3

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.*

1 Concurrent Systems

(*a*) A software module controls a car park of known capacity. Calls to the module's procedures *enter*() and *exit*() are triggered when cars enter and leave via the barriers.

Give pseudocode for the *enter* and *exit* procedures

    (*i*)   if the module is a monitor   [8 marks]

    (*ii*) if the programming language in which the module is written provides only semaphores   [4 marks]

(*b*) Outline the implementation of

    (*i*)   semaphores   [4 marks]

    (*ii*) monitors   [4 marks]

[TURN OVER

2 Further Java

Write brief notes on the facilities for lightweight concurrency in Java. You should cover the following topics, using code fragments to illustrate your answer:

(*a*) creating a thread

(*b*) mutual exclusion

(*c*) signalling

(*d*) asynchronous interruption

(*e*) managing priority

[4 marks each]

3 Compiler Construction

With reference to a strictly-typed block-structured programming language, write brief notes on the following topics:

(*a*) the allocation and recovery of records stored in a heap

(*b*) the implementation of variables of union type

(*c*) the allocation of arrays with non-manifest bounds

(*d*) the implementation of labels and GOTO commands

[5 marks each]

4 Introduction to Security

Consider the following two separation-of-duty policies:

(*a*) A transaction needs approval from two people, one in group A and one in group B.

(*b*) A transaction needs approval from two distinct users of the system.

Which of these is harder to implement using the standard Unix access control mechanisms, and why? [10 marks]

Sketch an implementation of the easier policy using Unix mechanisms. [5 marks]

Describe at least *two* alternative mechanisms that might be used to implement the other policy. [5 marks]

5 Data Structures and Algorithms

Describe an efficient algorithm based on Quicksort that will find the element of a set that would be at position k if the elements were sorted. [6 marks]

Describe another algorithm that will find the same element, but with a guaranteed worst case time of $O(n)$. [7 marks]

Give a rough estimate of the number of comparisons each of your methods would perform when $k = 50$, operating on a set of 100 random 32-bit integers. [7 marks]

6 Computer Design

Gordon Moore's law originally applied to memory size (in bits), but also applies to processor speed (in MIPS). However, main memory access latency does not follow Moore's law.

(*a*) What is Moore's law? [4 marks]

(*b*) Why doesn't main memory latency decrease in proportion to processor cycle time? [6 marks]

(*c*) What mechanisms are used to hide poor main memory access times and why do these mechanisms work? [10 marks]

[TURN OVER

7 Operating System Functions

Why are the scheduling algorithms used in general-purpose operating systems such as Unix and Windows NT not suitable for real-time systems?   [4 marks]

Rate monotonic (RM) and *earliest deadline first* (EDF) are two popular scheduling algorithms for real-time systems. Describe these algorithms, illustrating your answer by showing how each of them would schedule the following task set.

Task	Requires Exactly	Every
$\mathcal{A}$	2ms	10ms
$\mathcal{B}$	1ms	4ms
$\mathcal{C}$	1ms	5ms

You may assume that context switches are instantaneous.   [8 marks]

Exhibit a task set which is schedulable under EDF but not under RM. You should demonstrate that this is the case, and explain why.
[Hint: consider the relationship between task periods.]   [8 marks]

8 Continuous Mathematics

(*a*) When numerically computing the solution to an ordinary differential equation (ODE) that involves higher-than first-order derivatives:

  (*i*) What is to be done about the higher-than first-order terms, and how can this be accomplished?   [4 marks]

  (*ii*) Illustrate this step for the following ODE, in which functions $r(x)$ and $q(x)$ are known and we seek to compute the solution $y(x)$:

$$\frac{d^2y}{dx^2} + q(x)\frac{dy}{dx} = r(x) \qquad \text{[4 marks]}$$

(*b*) (*i*) State the incrementing rule for the Euler method of numerical integration, in terms of:

  - $f(x_n)$, the estimate of the solution $f(x)$ at the current point x_n

  - $f(x_{n+1})$, the new estimate of $f(x)$ for the next point x_{n+1}

  - the integration stepsize h, which is the interval $(x_{n+1} - x_n)$

  - $f'(x_n)$, the expression given by the ODE for the derivative of the desired solution $f(x)$ at the current point x_n

                                                                                                            [4 marks]

  (*ii*) What might happen to your solution if the stepsize h is too large?
                                                                                                            [2 marks]

  (*iii*) What might happen to your solution if you make the stepsize h too small?
                                                                                                            [2 marks]

  (*iv*) What is the primary advantage of the Runge–Kutta method over the Euler method for numerical integration of ODEs?   [2 marks]

  (*v*) Under what conditions might you wish to make the stepsize h adaptive rather than fixed? How should you adapt it?   [2 marks]

9 Computation Theory

One of the most important contributions of the theory of computation has been to establish that the halting problem is not decidable. Give a clear statement of this result (you are not asked to prove it). [5 marks]

Define a *configuration* of a 2-register machine at a particular point during the execution of some program. [3 marks]

By considering the total number of configurations or otherwise, show that it is not possible to compute an upper bound for the contents of the two registers during halting computations as a function of the program code and the initial contents of the two registers. [12 marks]

10   Numerical Analysis I

The parameters for IEEE Double Precision are: $\beta = 2$, $p = 53$, $e_{\min} = -1022$, $e_{\max} = 1023$. Explain the terms *significand*, *sign bit*, *exponent*, *normalised number*, *denormal number*, *hidden bit*, *precision* as used in IEEE arithmetic. What values does the hidden bit have for normalised and denormal numbers?   [8 marks]

In which order are the significand, sign bit and exponent stored? How is the exponent stored? Deduce how many bits are required to store the Double Precision exponent. How many bits are required to store a Double Precision number?

[5 marks]

Deduce the meaning of the following Double Precision bit patterns (where dots indicate a number of zeros):

(*a*)   01111111111100...00

(*b*)   11000000000000...00

(*c*)   01111111111110...00

(*d*)   11111111111100...01

(*e*)   10000000000000...00

(*f*)   00000000000010...00

(*g*)   00111111111100...00

[7 marks]

END OF PAPER