

COMPUTER SCIENCE TRIPOS Part IB

Monday 31 May 2004 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.*

**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 Computer Design

TLBs and caches are examples of content-addressable memories (CAMs).

- (a) What is the principal difference between a CAM and a RAM? [4 marks]
- (b) What is the difference between fully associative, set associative and direct mapped lookup? [6 marks]
- (c) Why are TLBs always much smaller than caches? [4 marks]
- (d) Which of the lookup mechanisms in part (b) is usually used for a TLB and why aren't the other mechanisms usually used? [6 marks]

2 ECAD

Consider the following mysterious Verilog module.

```

module mystery(c,r,a,s);

input  c,r,a;
output [2:0] s;
reg    [2:0] s;

always @(posedge c or posedge r)
  if(r)
    s<=0;
  else begin
    if(a && (s<7)) s<=s+1;
    else if(!a && (s>0)) s<=s-1;
  end
endmodule

```

- (a) How many flip-flops will be required to implement the mystery module, and how will signals `c` and `r` be connected to these flip-flops? [5 marks]
- (b) What is the state transition diagram for this mystery module? [5 marks]
- (c) If this module were synthesised to the minimum sum of products form, what would the equations be for next state bits `s[0]`, `s[1]` and `s[2]`? [10 marks]

3 Data Structures and Algorithms

- (a) Describe how the Lempel Ziv text compression algorithm works, illustrating your answer by deriving the sequence of numbers and corresponding bit patterns it would generate when applied to a string starting with the following 24 characters:

ABCDABCDABCDABCDABCDABCD . . .

You may assume that the initial table is of size 256 (containing bytes 0 to 255) and that the codes for “A”, “B”, “C” and “D” are 65, 66, 67 and 68, respectively. [12 marks]

- (b) Estimate how many bits the algorithm would use to encode a string consisting of 1000 repetitions of the character “A”. [8 marks]

4 Comparative Programming Languages

- (a) Discuss to what extent a programmer can expect a program that conforms to a standard to generate identical results when run under different conforming compilers on different machines. [6 marks]
- (b) ALGOL 60 provided call by value and call by name, Pascal provided call by value and call by reference, and ALGOL-W provided a variety of calling methods including call by result and call by value-result. Briefly describe the calling mechanisms just mentioned and discuss why most modern programming languages provide only call by value. [8 marks]
- (c) Discuss the reasons why languages such as Fortran, Algol and PL/I designed in 1950s and 1960s are less widely used than languages designed in the last 20 years. [6 marks]

5 Operating Systems II

- (a) What problem do real-time scheduling algorithms try to solve? [2 marks]
- (b) Describe one *static priority* and one *dynamic priority* real-time scheduling algorithm. You should discuss the issue of admission control, and comment on the data structures that an implementation would need to maintain and on how these would be used to make scheduling decisions. [8 marks]
- (c) A designer of a real-time system wishes to have concurrently executing tasks share a data structure protected by a mutual exclusion lock.
- (i) What scheduling problem could arise here? [2 marks]
- (ii) How could this problem be overcome? [2 marks]
- (d) The designer also wishes the real-time system to use demand paged virtual memory for efficiency. What problems could arise here, and how could they be overcome? [6 marks]

6 Numerical Analysis I

- (a) For Single Precision in the *IEEE* binary floating-point standard (*IEEE* 754) the *precision* is defined as 24, and the *exponent* requires 8 bits of storage. With reference to *IEEE* Single Precision, explain the terms *exponent*, *significand*, *precision*, *sign bit*, *normalised number*, *denormal number*. [6 marks]
- (b) Explain the term *hidden bit*. What are the values of the hidden bit for normalised and denormal numbers? How is the exponent stored and why? How are the exponent, significand and sign bit arranged in memory? [4 marks]
- (c) Let x^* denote the floating-point representation of a number x . Define the terms *absolute error* (ε_x) and *relative error* (δ_x) in representing x . How are ε_x and δ_x related? Define *machine epsilon* (ε_m). [3 marks]
- (d) Assume $\delta_x = \delta_y = \delta_z = \varepsilon_m$. Using worst-case analysis, estimate δ_{xy} , ε_{xy} . Find an expression for δ_w where $w = z - xy$. [4 marks]
- (e) Working to 4 significant decimal digits only, compute w^* when $x^* = 2.018$, $y^* = 2.008$, $z^* = 4.058$. Given $\varepsilon_m \simeq 0.5 \times 10^{-3}$, how many significant decimal digits of w^* can be relied on? [3 marks]

7 Computation Theory

- (a) What is Turing's Thesis? [2 marks]
- (b) Explain the action of a Turing machine that is specified by a quintuplet description. [4 marks]
- (c) Define the *configuration* of a Turing machine at step t , and establish equations that specify the configuration of a k -symbol Turing machine at step $(t + 1)$ in terms of the configuration at the previous step t . [6 marks]
- (d) Explain how you would use your equations to simulate a specific Turing machine by a register machine whose program encodes the quintuplet description. To what extent does this support Turing's Thesis? [Explicit program for a register machine is *not* required.] [8 marks]

8 Computer Graphics and Image Processing

Describe an algorithm for performing scan conversion of a set of 3D polygons, including details of clipping, projection, and the underlying 2D polygon scan conversion algorithm. You may assume that you are given the colour of each polygon and that no lighting calculations are required. Please state any additional assumptions that you need to make. *Ray tracing* is not an acceptable answer to this question. [20 marks]

9 Introduction to Security

- (a) Explain briefly mechanisms that software on a desktop computer can use to securely generate secret keys for use in cryptographic protocols. [5 marks]
- (b) Give *two* different ways of implementing residual information protection in an operating system and explain the threat addressed by each. [5 marks]
- (c) Consider the standard POSIX file-system access control mechanism:
- (i) Under which conditions can files and subdirectories be removed from a parent directory? [2 marks]
 - (ii) Many Unix variants implement an extension known as the “sticky bit”. What is its function? [2 marks]
 - (iii) On a POSIX system that lacks support for the “sticky bit”, how could you achieve an equivalent effect? [2 marks]
- (d) VerySafe Ltd offer two vaults with electronic locks. They open only after the correct decimal code has been entered. The VS100 – a low-cost civilian model – expects a 6-digit code. After all six digits have been entered, it will either open or will signal that the code was wrong and ask for another try. The VS110 – a far more expensive government version – expects a 40-digit code. Users of a beta-test version of the VS110 complained about the difficulty of entering such a long code correctly. The manufacturer therefore made a last-minute modification. After every five digits, the VS110 now either confirms that the code has been entered correctly so far, or it asks for the previous five digits again. Compare the security of the VS100 and VS110. [4 marks]

END OF PAPER