COMPUTER SCIENCE TRIPOS Part IB

Monday 5 June 2006   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

STATIONERY REQUIREMENTS
Script Paper
Blue Coversheets
Tags

1 ECAD

(*a*) In the Verilog hardware description language, what is the difference between *continuous*, *blocking* and *non-blocking* assignments? [6 marks]

(*b*) What is the labelled circuit diagram which corresponds to the following snippet of Verilog? [3 marks]

```
reg  [1:0] s;
wire t;
always @(posedge clk) s <= {s[0],t};
```

(*c*) A naïve programmer has written the following Verilog module to sort two 4-bit values `a` and `b` such that `a<b`.

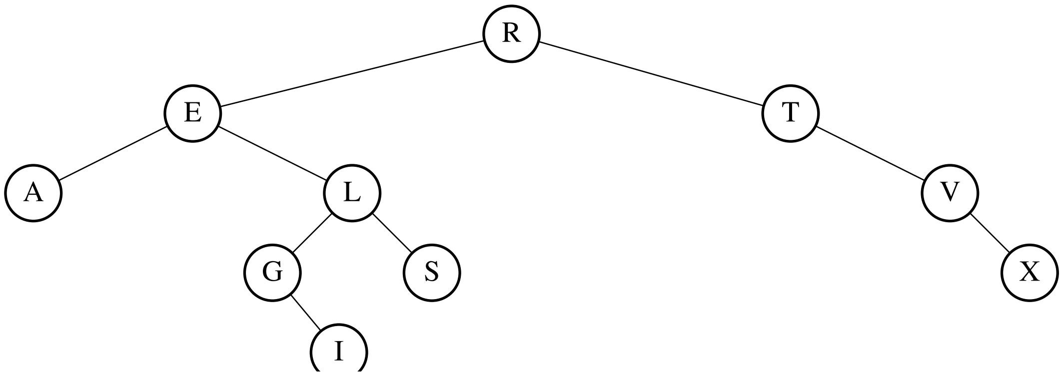
```
module sort2(clk,a,b);
  input clk;
  input [4:0] a;
  input [4:0] b;
  wire  [4:0] t;
  always @(clk) begin
    if(a>b) begin
      t <= a;
      a <= b;
      b <= t;
    end
  end
endmodule
```

(*i*) What is wrong with the code above? [6 marks]

(*ii*) How would you correct this module so that it takes two values as input and outputs the sorted values one clock cycle later? [5 marks]

2 Data Structures and Algorithms

(*a*) Briefly explain what a binary search tree (BST) is, listing its properties. Is the following binary tree a BST or not, and why?



[3 marks]

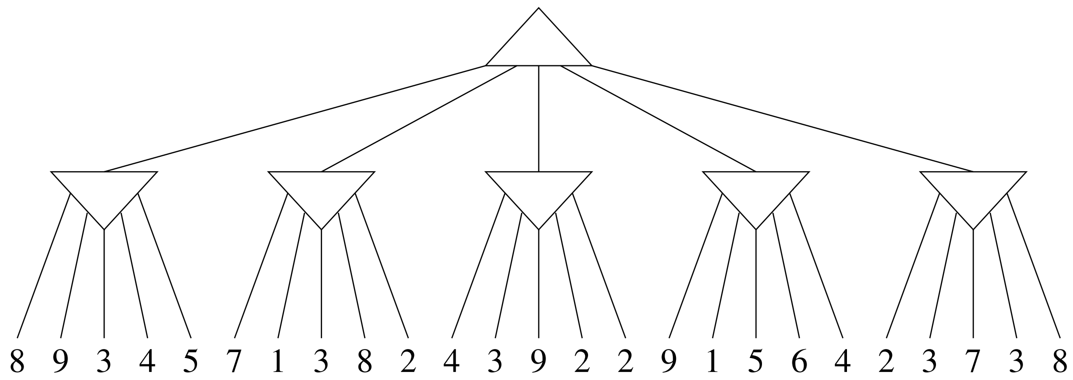
(*b*) Describe an optimally efficient algorithm to find the predecessor of a given node n in a BST and explain why it works. [6 marks]

(*c*) Describe an optimally efficient algorithm for deleting a node d from a BST when neither of d's subtrees is empty. Explain why it works and prove that what remains is still a BST. [5 marks]

(*d*) Assume that node l, whose key is k_l, is a leaf of a BST and that its parent is node p, with key k_p. Prove that, of all the keys in the BST, k_p is either the smallest key greater than k_l or the largest key smaller than k_l. [6 marks]

3 Artificial Intelligence I

(*a*) Give a detailed description of the *minimax algorithm* for two-player games, illustrating your answer using the following game tree.



[10 marks]

(*b*) Describe the modifications required to the minimax algorithm in order to apply it to realistic games. [5 marks]

(*c*) Give a detailed description of the technique of $\alpha - \beta$ pruning, again illustrating your answer using the game tree above. [5 marks]

4 Comparative Programming Languages

(*a*) In order to remove the overhead of a function call, a programmer decides to replace all calls to a function `f` with the macro `F`, where `f` and `F` are defined as follows:

```
int f(int x) { return x+x;}

#define F(X) (X)+(X)
```

(*i*) Give two valid C expressions involving `f` which produce different results when `F` is substituted for `f`. Justify your answer. [4 marks]

(*ii*) State the C language feature which can be used to correctly remove the overhead of a function call. [1 mark]

(*b*) Consider the following:

```
static struct link {
  int v;
  struct link *next;
} *head=0;

void convert(int a[], int len);
```

Write a function definition for `convert` which updates `head` to point to a linked-list containing the elements of `a` in the same order. You may assume `len` contains the number of elements in `a`. [5 marks]

(*c*) Consider the following C++ declaration:

```
template<int n> int SumSquares();
```

(*i*) Using function specialisation, provide an implementation of `SumSquares` so that, given an integer `N`, `SumSquares<N>()` returns:

$$\sum_{i=1}^{\mathtt{N}} i^2$$

[5 marks]

(*ii*) Compare and contrast the functionality of the C preprocessor and the C++ template system. Explain why it is not possible to write a C preprocessor macro to implement `SumSquares`. [5 marks]

5 Operating Systems II

Most modern computers include a *translation lookaside buffer* (TLB) to speed up address translation.

(*a*) Describe with the aid of a diagram the basic operation of a TLB. [2 marks]

(*b*) Some TLBs support *process tags* (sometimes called *address space numbers*). Explain how process tags could be used by an operating system, and what benefits might be expected. [2 marks]

(*c*) Some TLBs support *superpages* of one or more sizes.

(*i*) Explain how superpages could be used to reduce the TLB footprint for operating system kernels. [2 marks]

(*ii*) Explain how superpages could be used to reduce the TLB footprint for processes. What additional considerations need to be taken into account in this case? How well do you think your scheme would work in practice? [4 marks]

(*d*) Compare and contrast the way in which demand paging is performed in Unix and VMS. [10 marks]

6 Numerical Analysis I

(*a*) An IEEE Single Precision number is stored in 32 bits, of which 8 bits are reserved for the exponent. Explain the terms *normalised number* and *hidden bit*. How many bits are used to store the *significand*, and what is the *precision*? Show by means of a diagram how the bits are arranged in storage. [4 marks]

(*b*) How is the value of the exponent stored? What are the stored values of the exponents e_{min} and e_{max}? [3 marks]

(*c*) Which values are represented by the following bit patterns? [Show signs where appropriate.]

(*i*) 00000000 00000000 00000000 00000000

(*ii*) 11111111 11111111 11111111 11111111

(*iii*) 00111111 10000000 00000000 00000000

(*iv*) 11000000 00000000 00000000 00000000

(*v*) 00000000 00000000 00000000 00000001

(*vi*) 01111111 10000000 00000000 00000000 [6 marks]

(*d*) Define *machine epsilon* ε_m. Estimate its value in IEEE Single Precision. [2 marks]

(*e*) What are the two sources of error in the formula

$$f'(x) \simeq \frac{f(x+h) - f(x)}{h}$$

and how does each type of error behave as h increases? Suggest a suitable value of h to use with this formula for IEEE Single Precision when $f(x) = O(1)$. [5 marks]

7 Computation Theory

(*a*) (*i*) Give a graphical representation of the following register machine program.

$$
\begin{aligned}
L0 &: \quad Z^+ \to L1 \\
L1 &: \quad L^- \to L2, L3 \\
L2 &: \quad Z^+ \to L0 \\
L3 &: \quad Z^- \to L4, L5 \\
L4 &: \quad L^+ \to L3 \\
L5 &: \quad X^- \to L1, L6 \\
L6 &: \quad \text{HALT}
\end{aligned}
$$

[3 marks]

(*ii*) Assuming the contents of register Z is initially 0, when the program is run starting at instruction $L0$ what functions of the initial contents of registers X and L are computed in X and L when the machine halts? [5 marks]

(*b*) (*i*) What is meant by a *Turing machine*, its *configurations*, *transition relation* and the *computations* it carries out? What does it mean to say that a computation *halts*? [6 marks]

(*ii*) Given a Turing machine, is it decidable whether or not for all possible initial configurations the machine will not halt after 100 steps of transition? Justify your answer. [6 marks]

8 Computer Graphics and Image Processing

(*a*) Describe, in outline, each of the z-buffer, BSP tree, and painter's algorithm methods for rendering a set of 3D polygons. [4 marks each]

(*b*) Compare and contrast the three methods. [8 marks]

9 Introduction to Security

(*a*) Name *three* types of software vulnerability; give an example of each and a brief description of how each could be exploited. [9 marks]

(*b*) Alice wants to attack Bob's computer via the Internet, by sending IP packets to it, directly from her own computer. She does not want Bob to find out the IP address of her computer.

(*i*) Is this easier to achieve for Alice with TCP- or UDP-based application protocols? Explain why. [3 marks]

(*ii*) For the more difficult protocol, explain *one* technique that Alice could try to overcome this obstacle and *one* countermeasure that Bob could implement in his computer. [3 marks]

(*iii*) Name *three* functions that Alice's Internet service provider could implement to make it more difficult for Alice to achieve her goal? [3 marks]

(*c*) In what way are TCP/UDP port numbers below 1024 special? [2 marks]

10 Mathematical Methods for Computer Science

(*a*) Suppose that $X_1, X_2, \ldots$ is a sequence of random variables. State the Central Limit Theorem, noting any assumptions that you make about the random variables, X_i, and stating carefully the type of limiting convergence that is being considered. [2 marks]

(*b*) Suppose that $X_1, X_2, \ldots, X_n$ is a random sample from a population with unknown mean μ.

(*i*) Explain what is meant by a $100(1-\alpha)$ percent confidence interval for the unknown parameter μ. [2 marks]

(*ii*) Derive an approximate $100(1-\alpha)$ percent confidence interval for μ using the Central Limit Theorem. [4 marks]

(*iii*) Apply your method to construct an approximate 95% confidence interval for the population mean, μ, given the following random sample of 20 values taken from the population: 10.583, 6.775, 12.126, 9.135, 12.690, 10.638, 11.683, 10.428, 11.201, 7.911, 10.608, 9.477, 9.643, 8.263, 13.909, 8.563, 10.910, 9.962, 14.108, 11.201. [4 marks]

(*c*) Lift A has a sign reading "`13 people, 1000 kg`" and lift B has a similar sign reading "`4500lb or 30 persons`". Given a standard deviation of $\sigma = 20$ `kg` for the weight of a person, what is an approximate mean weight of a person in `kg` such that the probability a group of maximum size exceeds the weight limit of a lift with probability p? Apply your method in the cases of lifts A and B separately and with the values of $p = 0.1$ and $p = 0.01$. You may use the approximation that 1 `kg` $\approx$ 2.2 `lb`. [8 marks]

The following table gives values of $\Phi(x) = P(Z \leq x)$, the distribution function of Z, where Z is a random variable with the standard normal distribution.

x	0.842	1.282	1.645	1.96	2.326
$\Phi(x)$	0.8	0.9	0.95	0.975	0.99

END OF PAPER