Birkbeck
(University of London)

MSc Examination for Internal Students

School of Computer Science and Information Systems

Fundamentals of Computing (COIY058H7)
Credit Value: 15

Date of Examination: Friday 22 May 2009
Time of Examination: 10:00 – 12:00
Duration of Paper: Two hours

This paper is split into ***two*** *sections.*
There are 50 marks for each section.
There are 5 questions in Section A and 6 questions in Section B.

Start each section in a separate answer book.

Answer all questions.

The use of electronic calculators is not permitted.

The breakdown of marks per question is as follows:

Question	1	2	3	4	5	6	7	8	9	10	11
Marks	10	10	10	10	10	7	8	7	8	10	10

Section A

1. Construct the truth-table for the Boolean function that is realised by the Boolean formula

$$B \;=\; (A_1 \to \neg A_3) \vee (A_3 \to A_2).$$

Use the truth-table to realise this function by a formula with the connectives $\neg$ and $\vee$ only. Simplify the formula in such a way that the corresponding Boolean circuit contains at most four gates. Show the Boolean circuit. (10 marks)

2. Represent the number -18 as a two's complement 32-bit binary number and as an IEEE 754 32-bit floating-point number. Show your working. (10 marks)

3. Design a (deterministic or nondeterministic) finite automaton A such that $L(A)$ consists of all strings over the alphabet $\{0, 1\}$ that begin with 01 and contain an odd number of 0s. (10 marks)

4. Let $\Sigma = \{a, b\}$. For each of the following languages over Σ, find a regular expression representing it: (10 marks)

   (i) All strings that contain two or three bs.

   (ii) All strings that do not contain ab as a substring.

5. Consider the following transition table of a Turing machine (with s being its initial state):

s	$\sqcup$	h	$\sqcup$
s	1	q	$\rightarrow$
s	$\rhd$	q	$\rightarrow$
q	$\sqcup$	p	1
q	1	s	$\rightarrow$
q	$\rhd$	q	$\rightarrow$
p	$\sqcup$	h	1
p	1	p	$\rightarrow$
p	$\rhd$	s	$\rightarrow$

Give the computations of the machine on inputs 1, 1111 and 111. Describe in English what this Turing machine does. (10 marks)

Section B

6. A queue is represented by a *singly-linked circular* list with a list *head*. Each node has two fields, **INFO** and **LINK**, and the **LINK** field of the rear node of the queue points to the head node. In the head node the **LINK** field points to the front node of the queue and the **INFO** field (which in the head node is a pointer) points to the rear node of the queue, but in an empty queue they both point to the head node itself.

   (a) Draw the list after A, B, C and D have been inserted into the (originally empty) queue in that order. (1 mark)

   (b) Write algorithms for the queue operations **Queue(HEAD, y)**, which inserts the data value **y** into the queue, and **Unqueue(HEAD)**, which removes an item from the queue and returns its value. (6 marks)

7. (a) Draw the *binary search tree* obtained by inserting the following three-letter strings into an empty tree in the given order: (2 marks)

   hog, *yak*, *fox*, *pig*, *owl*, *cat*, *emu*, *fly*, *rat*, *bee*.

   (b) Write down the strings in the order the nodes are visited if the tree is traversed in (i) *preorder*, (ii) *postorder*. (3 marks)

   (c) Add in the threads to turn your tree into a *threaded* binary tree with a *head* node. (1 mark)

   (d) Draw the *forest* corresponding to your binary tree using the *natural correspondence* between forests and binary trees. (2 marks)

8. Consider the following algorithm, where `ROOT` is a pointer to the root node of a binary tree:

```
PointerStack S
Clear(S)
Stack(ROOT)
while (not Empty(S))
{      P ← Unstack(S)
       Visit(P)
       if (P↑Llink ≠ nil) Stack(P↑Llink)
       if (P↑Rlink ≠ nil) Stack(P↑Rlink)
}
```

   (a) Trace the execution of this algorithm on the binary search tree you constructed in the previous question (i.e., 7(a)), and write down the order in which the nodes are visited. (6 marks)

   (b) State any relationship that this order has to any of the standard traversal orders. (1 mark)

9. (a) If P is a *leaf* node in a binary tree and Q is the parent of P, is it always the case that either P is the inorder successor of Q or Q is the inorder successor of P? Justify your answer. (4 marks)

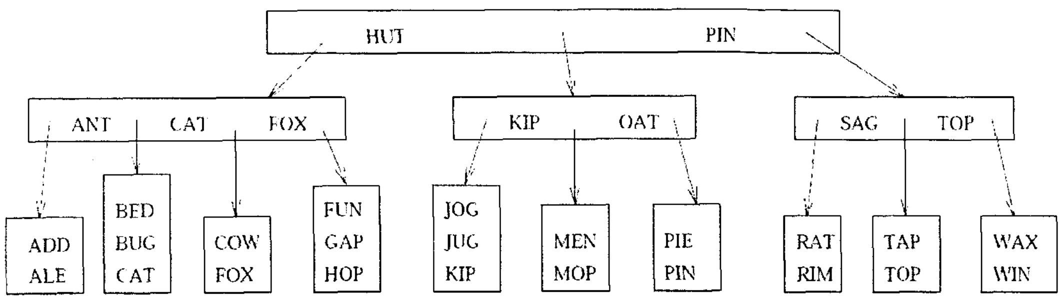
(b) Give an algorithm for the function **Parent(P)** that returns a pointer to the parent of any *leaf* node **P** in a *threaded* binary tree. (4 marks)

10. (a) The following diagram shows a B-tree of order 5 with a maximum of 3 records per data page. Starting in each case with the B-tree shown in the diagram, consider the *deletion* of a single record from the B-tree. Give one example for which, in the resulting B-tree, the total number of pages is reduced by *one* and one example for which it is reduced by *two*. Draw the resulting B-tree in each case. (8 marks)

Notes:

i. When combining pages, a page should always be combined with its sibling page to its *left* (unless it does not have one).

ii. You do *not* need to write in the contents of any *data* pages that are unchanged.



(b) What is the smallest number of records that would have to be deleted from the B-tree shown that could cause the number of levels in the B-tree to decrease from 3 to 2? (2 marks)

[Do *not* state which records would have to be deleted or draw the resulting B-tree.]

11. (a) Use the divide-and-conquer strategy to write a function:

$$\texttt{MaxMin(j, k, max, min)}$$

that finds the maximum and minimum values in the integer array

$$\texttt{a[j], a[j + 1], ..., a[k]}$$

and assigns them to the reference parameters `max` and `min`, respectively. (6 marks)

Note: If there are only two elements, i.e., j + 1 = k, you only have to make one comparison between the two elements.

(b) $T(n)$ denotes the number of comparisons of array elements made by the function when the array contains n elements; so clearly

$$T(1) = 0 \qquad \text{and} \qquad T(2) = 1.$$

Write down a recurrence relation (equation) for $T(n)$.
You may assume that n is a (positive) power of 2 and is greater than 2. (2 marks)

(c) Using your recurrence relation, calculate $T(n)$ for $n = 4, 8, 16, 32, 64$.
By examining the pattern of these values, suggest an expression for the value of $T(n)$ when n is any positive power of 2 greater than 1. (2 marks)