PAPER CODE NO.	EXAMINER : Prudence Wong
COMP108	DEPARTMENT : Computer Science Tel. No. 794-3694

THE UNIVERSITY
of LIVERPOOL

MAY 2005 EXAMINATIONS

Bachelor of Arts : Year 1
Bachelor of Science : Year 1
Master of Mathematics : Year 2
No qualification aimed for: Year 1

ALGORITHMIC FOUNDATIONS

TIME ALLOWED : TWO hours

INSTRUCTIONS TO CANDIDATES

Answer **four** questions only.

- Candidates will be assessed on their best four answers.
- If you attempt to answer more questions than the required number of questions, the marks awarded for the excess questions will be discarded (starting with your lowest mark).
- All logarithms are to the base 2.

THE UNIVERSITY
of LIVERPOOL

Question 1

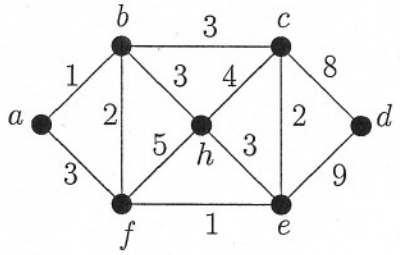
1A. Give the trace table and the output of the following algorithm for $m = 5$ and $n = 3$.

Input m, n
$i = 0$, $x = 1$, $s = 0$
while $i < n$ **do**
begin
$x = x * m$
$s = s + x$
$i = i + 1$
end
Output s

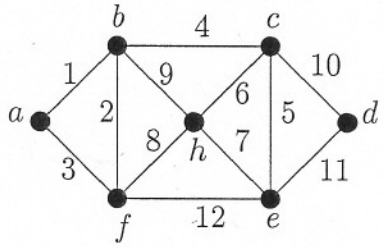
What is the output of this algorithm for general positive integers m and n?
What is the time complexity of this algorithm (in big-O notation)?

[10 marks]

1B. Consider the following graph. The label of an edge is the cost of the edge.



I. Give a table listing all edges and their costs in ascending order of the costs.
II. Using Kruskal's algorithm, draw a minimum spanning tree (MST) of the graph. Write down the order of the edges selected.
III. Is the MST drawn in (II) unique? (i.e., is it the one and only MST for the graph?)
IV. The graph below has a unique MST. Briefly explain why.



[15 marks]

Question 2

2A. Given a length-n sequence S of characters $S[0..(n-1)]$, a length-m sequence of characters $T[0..(m-1)]$ is called a *substring* of S if there exists some $0 \leq i \leq n - m$ such that $S[i..(i + m-1)] = T[0..(m-1)]$. T is further called a *unique* substring of S if T is a substring that appears exactly once in S. For example, if S is ACGTACGGG, then ACGG is a unique substring of S while neither ACG nor ACGA is a unique substring of S.

Design and write a pseudo code algorithm to determine if T is a unique substring of S.

What is the worst case time complexity of your algorithm (in big-O notation)?

[15 marks]

2B. Prove that the function $3n^2 \log n + 2n^2 + 5n + 1$ is $O(n^2 \log n)$.

[10 marks]

Question 3

3A. State (without proof) the least order of magnitude of the following functions.

I. $20 + 15n^3 + 10n^4 + 5n^2 + n^5$

II. $n^4 + n^3 \log^2 n + n^2 + 5$

III. $4n^2 \log^2 n + n^3 \log n + 5n^2 + n$

IV. $\sqrt{n} + \log n + 5$

V. $\sqrt{n^3} + n$

[10 marks]

3B. Prove the following statement by mathematical induction for all positive integers n.

$$\sum_{k=1}^{n}(2k+1) = n(n+2).$$

[10 marks]

3C. I. Describe what a *decision problem* is and what an *optimisation problem* is.

II. Classify the following problems based on whether they are a decision problem or an optimisation problem.

a. Given a graph G with integer weights on its edges. What is the weight of the shortest path between a node u and a node v?

b. Given a graph G with integer weights on its edges. Does G have a minimum spanning tree of weight at most 20?

III. Name one NP-complete problem.

[5 marks]

Question 4

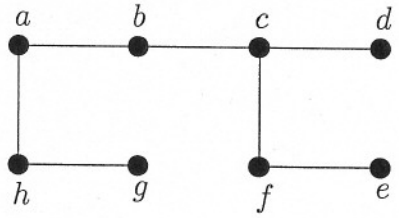
4A. Solve the following recurrence formula for any positive integer n by the *iterative method.*

$$T(n) = \begin{cases} 1 & \text{if } n = 1, \\ 3T(n-1)+1 & \text{if } n > 1. \end{cases}$$

(Hint: You can use the fact that $1 + x + x^2 + \ldots x^{n-1} = (x^n - 1)/(x - 1)$ for any positive integer n and any real number $x \neq 1$.)

[15 marks]

4B. Consider the following undirected graph.



I. What is the degree of the graph?

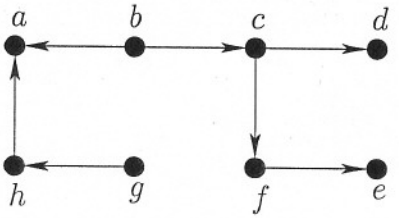
II. Starting from the node a, write down the vertices in the order of a depth first search (DFS) traversal.

III. Starting from the node a, write down the vertices in the order of a breath first search (BFS) traversal.

N.B.: There may be more than one solution for (II) and (III). You only need to put down one of them.

[5 marks]

4C. Consider the following directed graph.



I. What are the sum of in-degree and the sum of out-degree over all vertices for the above directed graph?

II. What is the relationship between the sum of in-degree, the sum of out-degree over all vertices, and the number of edges in a directed graph?

[5 marks]

Question 5

5A. Consider the recurrence formula

$$f(n) = \begin{cases} 1 & \text{if } n = 0 \text{ or } n = 1, \\ 2f(n-1) + 3f(n-2) & \text{if } n > 1. \end{cases}$$

I. Design and write the pseudo code of a recursive algorithm to compute the value $f(n)$.

II. The algorithm in (I) is not efficient. Design and write the pseudo code of a faster (non-recursive) algorithm using the concept of dynamic programming. What is the time complexity of the faster algorithm (in big-O notation)?

[15 marks]

5B. I. Recall that the selection sort algorithm sorts a sequence of numbers as follows: pick the smallest number and swap with the first number in the sequence, pick the second smallest number and swap with the second number in the sequence, and so on. Complete the trace table of the selection sort algorithm on sorting the array $a[\,]$ containing the numbers $4, 5, 2, 3, 1$.

	$a[0]$	$a[1]$	$a[2]$	a[3]	a[4]
Initial	4	5	2	3	1
1st round	⋮	⋮	⋮	⋮	⋮
⋮	⋮	⋮	⋮	⋮	⋮
Final	1	2	3	4	5

II. What is the worst case time complexity of the selection sort algorithm (in big-O notation) to sort n numbers?

III. Name two more sorting algorithms. At least one of them must have a different time complexity from the selection sort algorithm.

Give their worst case time complexities (in big-O notation).

[10 marks]