THE UNIVERSITY OF AUCKLAND

SECOND SEMESTER, 2005
Campus: City

COMPUTER SCIENCE

Algorithms and Data Structures

(Time allowed: TWO hours)

NOTE: Attempt all questions. Write answers in the boxes below the questions. You may use the "extra page" provided at the back if necessary for rough working only. Marks for each question are shown below the answer box. The use of calculators is NOT permitted.

SURNAME:

FORENAME(S):

STUDENT ID:

Section:	A	B	C	Total
Possible marks:	35	45	40	120
Awarded marks:				

QUESTION/ANSWER SHEETS FOLLOW

Surname: ______________________________

Forename(s): ______________________________

A. Algorithm Analysis

1. Match each algorithm or data structure with its description. Write the letter corresponding to the algorithm/data structure in the box next to the description. Algorithms/data structures are:

A. quicksort	B. quickselect	C. median-of-3 pivoting strategy
D. insertion sort	E. selection sort	F. open addressing with double hashing
G. Shellsort	H. mergesort	I. interpolation search
J. hash table	K. sequential search	L. open addressing with linear probing
M. heapsort	N. maximum heap	O. binary search tree

Can easily become unbalanced, hence the need for rotations.	
Runs in linear time on average, quadratic in worst case.	
Runs in $O(n \log n)$ time on average, quadratic in worst case.	
First nontrivial algorithm analysed by Knuth.	
Often finds an entry in a sorted list in $O(\log \log n)$ time.	
Reduces likelihood of quadratic performance of quicksort.	
A simple algorithm whose analysis is very difficult.	
Best sorting method if swaps are VERY expensive.	
Search operation is faster on average than for AVL trees.	
The only choice for searching linked lists.	
A clever array implementation of the priority queue ADT.	
In-place, stable, quadratic sorting method.	
Less prone to primary clustering.	
A stable $n \log n$ sorting method.	
The best sorting choice when speed is essential and storage is VERY small.	

[15 marks]

2. The following questions concern nonnegative functions f, g defined on the natural numbers. Answer each question true (T) or false (F).

   (a) If $f(n) = n^2, g(n) = (1 + (-1)^n)n$, then g is $O(f)$.

   (b) If $f(n) = n, g(n) = (1 + (-1)^n)n^2$, then g is $\Omega(f)$.

   (c) If $f(n) = n^2 \log_4 n, g(n) = (5n^2 + 1)(\lg n + \lg \lg n)$ then f is $\Theta(g)$.

   (d) There exist f and g such that f is $O(g)$ and g is $O(f)$ but f is not $\Theta(g)$.

   (e) $f + g$ is $\Theta(\max\{f, g\})$.

[5 marks]

Surname: ______________________________

Forename(s): ______________________________

3. In the textbook it is shown that the average running time $T(n)$ for quicksort satisfies the recurrence

$$T(n) = (2/n) \sum_{i=0}^{n-1} T(i) + cn, \text{ for } n \geq 1,$$

while the analogous quantity $U(n)$ for quickselect satisfies

$$T(n) = (1/n) \sum_{i=0}^{n-1} T(i) + cn, \text{ for } n \geq 1,$$

(a) Solve each of the recurrences above explicitly assuming $T(0) = U(0) = 0$.

[10 marks]

Surname: ______________________________

Forename(s): ______________________________

(b) Suppose we have to find 100 different order statistics from an array of size 10^6. Which is better: running quicksort and then fetching the desired elements, or running quickselect 100 times? Assume that fetching an array element takes zero time.
Assume that fetching an array element takes zero time. You may find the approximation $\Sigma_{k=1}^{n} 1/k \sim \ln n$ useful.

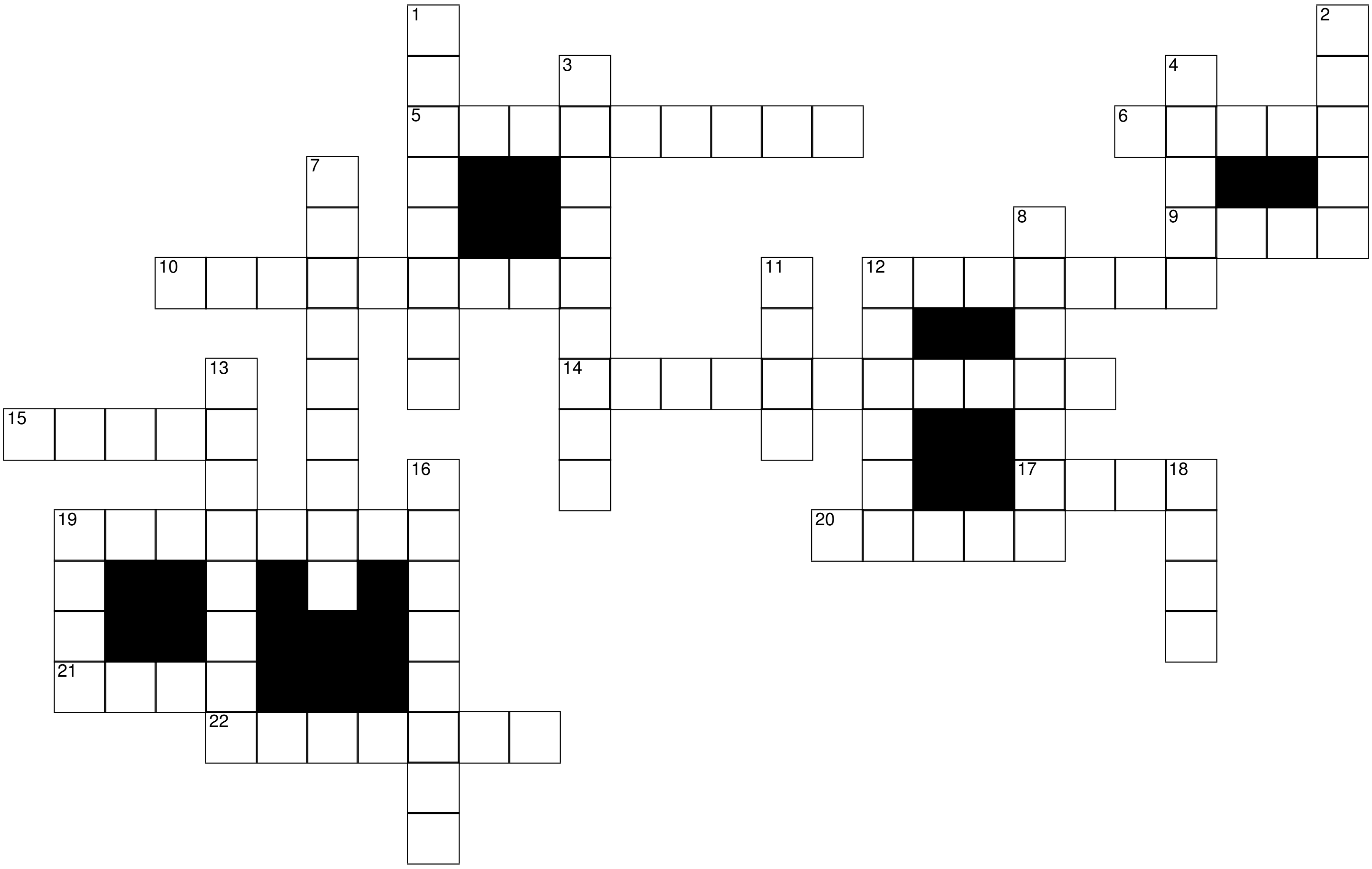
[5 marks]

Surname: ______________________________________

Forename(s): ______________________________________

B. Graph Algorithms

4. Solve the following crossword puzzle on graph terminology.



Across. 5 procedure to compute something 6 number of nodes 9 sequence of unique nodes following arcs 10 for graphs, equivalent to 2-colorable 12 ADT with nodes and arcs 14 type of order for nodes of a DAG 15 closed walk with distinct arcs 17 connected graph without cycles 19 subset of a graph that is also a graph 20 type of search yielding no cross edges 21 connects vertices 22 first 220 textbook author

Down. 1 maximum of minimum path lengths 2 smallest cycle length 3 having path between every pair of vertices 4 ADT with vertices and edges 7 systematic way to visit each node 8 type of search used to find diameter 11 sequence of adjacent nodes 12 number of neighbors 13 assigning numbers to edges 16 Dijkstra's paths 18 sum of all degrees 19 number of arcs

[15 marks]

Surname: ______________________________

Forename(s): ______________________________

5. Draw examples that satisfy the following properties.

   (a) A strongly connected digraph with order 4 and size 5.

   [5 marks]

   (b) An edge-weighted graph of order 5, size 8, and exactly two different minimum spanning trees of total weight 10. (Highlight the two trees in your answer.)

   [5 marks]

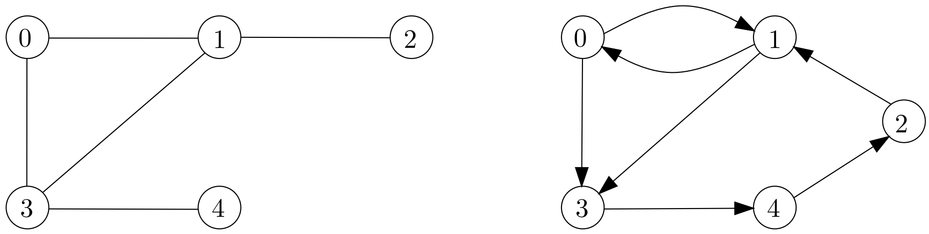
   (c) Two different graphs, both of order 6, size 7 and with the same vertex degrees, but only one graph is 2-colorable. (Explain why.)

   [5 marks]

Surname: ______________________________

Forename(s): ______________________________

6. For the following graph G_1 and digraph G_2 answer the following questions.



(a) Give adjacency lists for G_1 and adjacency matrix for G_2.

[5 marks]

(b) Do a DFS starting at node 0, indicating seen/done times on your trees.

[5 marks]

Surname: ______________________

Forename(s): ______________________

(c) Give the distance matrix (all-pairs shortest paths) for G_2, assuming all edge weights are equal to 1.

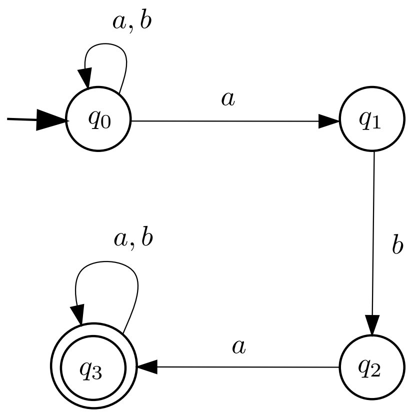
[5 marks]

Surname: ________________________________

Forename(s): ________________________________

C. Automata and Formal Languages

7. Consider the following NFA N:



(a) Find the minimal (in length) string $x \in \{a, b\}^*$ accepted by N and write a trace of computation which accepts x in N.

[4 marks]

(b) Find the minimal (in length) string $x \in \{a, b\}^*$ not accepted by N.

[4 marks]

Surname: ______________________________

Forename(s): ______________________________

(c) Determine the language L accepted by N.

[5 marks]

(d) Write a regular expression denoting L.

[5 marks]

8. Write an algorithm which receives as input a DFA M over the alphabet $\{a, b\}$ and decides whether $L(M) = \{a\}$ or $L(M) \neq \{a\}$. Clearly state all results you use.

[10 marks]

Surname: ______________________

Forename(s): ______________________

9. Let $L = \{w \in \{a, b\}^* \mid |w| \geq 2\}$.

   (a) Construct a DFA M accepting L.

   [5 marks]

   (b) Is the DFA M minimal? If yes, justify your claim; if no, construct a minimal DFA equivalent to M.

   [5 marks]

   (c) Construct a DFA accepting the complement of L.

   [2 marks]