OLLSCOIL NA hÉIREANN
THE NATIONAL UNIVERSITY OF IRELAND

NATIONAL UNIVERSITY OF IRELAND, GALWAY

SUMMER EXAMINATIONS 2001

Masters in Information Technology
Masters of Engineering Science

Algorithms
CT510

Prof. D. Bell
Prof. G. Lyons
Ms. J. Griffith

Time allowed: **Two hours**

Answer any THREE questions
All questions carry equal marks

1. i) Describe a queue data structure, discussing a suitable representation of the data structure. Using the representation of a queue presented, describe algorithms, and give pseudocode, for the insertion and deletion of an item.

ii) Describe a linked list data structure. Outline algorithms for inserting an item into a linked list and deleting an item from a linked list. Compare these insertion and deletion algorithms with algorithms for the same operations using an array data structure.

iii) Given two sorted arrays of length N and M, develop and write pseudocode for an algorithm which merges the two arrays into a single array of length N + M such that the sorted order is maintained.

2. i) What is meant by a *well-defined recursive procedure*? Outline a recursive algorithm to find the sum of the first *n* integer numbers, e.g.,

`sum(4)` returns `10`

ii) Outline a recursive and non-recursive algorithm to search for an item in an array where the elements in the array are sorted in non-decreasing order. Comment on the efficiency of the algorithms you develop. Compare the efficiency of searching an array, whose elements are in sorted order, with searching a linked list, whose elements are in sorted order.

iii) Describe Huffman's algorithm for compression. Use the algorithm described to build a Huffman tree given the following characters and the associated frequency of their occurrence in a file:

A	*B*	*C*	*D*	*E*	*F*	*G*	*H*	*Y*
22	*5*	*11*	*19*	*22*	*11*	*25*	*5*	*2*

List the resulting code for each letter.
What metrics are used to measure the efficiency of Huffman's algorithm for compression.

3. i) What is meant by *hashing*? With the aid of an example describe *any* hashing function.

ii) With respect to hashing techniques, explain what is meant by a *collision resolution technique*. By inserting the data (Key value) given in the example below (with associated hash address) describe the operation of the following collision resolution techniques:

- linear probing.
- chaining.

Key value:	A	B	C	D	E	X	Y	Z
Hash address:	3	7	1	10	3	10	4	0

(using an array of size 11, with locations from 0 to 10).

Outline the advantages and disadvantages of each technique.

iii) Give pseudocode functions for searching and insertion using hashing techniques. Choose any hashing function and any collision resolution technique, stating any assumptions made.

4. i) Describe a binary search tree data structure. Discuss a representation of a binary search tree. Given a binary search tree, T, write a procedure which finds the number of nodes in T. Explain your answer.

ii) With the aid of an example, outline the steps involved in deleting an item from a binary search tree. Include pseudocode.

iii) With the aid of an example, describe what is meant by a balanced binary search tree. Develop an algorithm which searches for an item in a binary search tree. Discuss the efficiency of the algorithm you develop and comment on how the efficiency is affected if the tree is unbalanced.

5. i) Distinguish between preorder, postorder and inorder traversal of a binary tree, giving pseudocode for *any one* of the traversal mechanisms. Comment on the results of inserting the following data into a binary search tree and then performing an inorder traversal of the binary search tree (show developed tree and results of traversal):

30 40 23 33 17 29 50 5

ii) With the use of sample data, describe how the selection sort algorithm operates. Give pseudocode for selection sort.

iii) Divide and conquer approaches to sorting achieve an improvement on selection sort.

a) Explain what is meant by a *divide and conquer* approach.
b) Outline a divide and conquer algorithm for sorting, including pseudocode in the answer.
c) Discuss the improvement on selection sort achieved by the divide and conquer sorting algorithm presented in part b).