NATIONAL UNIVERSITY OF SCIENCE AND TECHNOLOGY

FACULTY OF APPLIED SCIENCE

COMPUTER SCIENCE DEPARTMENT

AUGUST SUPPLEMENTS EXAMINATIONS 2004

SUBJECT: ALGORITHMS AND DATA STRUCTURES
CODE: SCS2103

INSTRUCTION TO CANDIDATES

Answer any 5 questions
Write all code in C

3 HOURS

QUESTION ONE

Write a C program to implement a linked circular queue. The queue should maintain details of incoming telephone calls. Each queue item should include a telephone number and a time of arrival. Include the following functions in your implementation

(a) enqueue, to add an item to the end of the queue [5]
(b) dequeue, to remove the first item from the queue [5]
(c) count, to return the number of items in the queue [5]
(d) display to display the items in the queue. [5]

QUESTION TWO

Write a C program, which is a dynamic tree arrangement. Each list item should include an integer, and the number of times that particular integer has been entered into the tree.

(a) Provide a suitable declaration for the `struct` that describes each tree node. [2]
(b) Provide a suitable declaration for the storage array. [1]

Implement the following functions:

(c) insert, to add an integer to the tree [5]
(d) count, to return the number of nodes in the tree [3]
(e) preorder, to perform a preorder traversal of the tree [3]
(f) inorder, to perform an inorder traversal of the tree [3]
(g) postorder, to perform a postorder traversal of the tree [3]

NB: Your visits to each node should print out the integer stored there, as well as the number of times each integer was inserted into the tree.

QUESTION THREE

Write a C function which appends a static list to another static list to produce a resulting list. The function should not destroy the original lists. [20]

QUESTION FOUR

Compare and contrast the time complexity of the following internal sorting algorithms using a model of execution of your choice and summarize using the Big-O notation.

Algorithm 1:

```
void straight_insertion(float a[], int n) {
      int i, j;
      T temp;

      for (i = 1; i < n; i++) {
            temp = a[i];
            j = i;
            while (temp < a[j-1]) {
                  a[j] = a[j-1];
                  j--;
            }
            a[j] = temp;
      }
}
```

and

Algorithm 2:

```
void bubble_sort(float a[], int n) {
      int i, j;
      T temp;

      for (i = 1; i < n; i++) {
            for (j = n-1; j >= i; j--) {
                  if (a[j-1] > a[j]) {
                        temp = a[j-1];
                        a[j-1] = a[j];
                        a[j] = temp;
                  }
            }
      }
}
```

[20]

QUESTION FIVE

Implement a dynamic stack of floats. Be sure to include the following operations:

(a) Push, to add an element to the top of the stack. [5]
(b) Top, to return the value stored at the top of the stack. [5]
(c) Pop, to remove and return the top-most value in the stack. [5]
(d) Height, to return the number of nodes in the stack. [5]

QUESTION SIX

(a) Write a C program to implement a linked list. Include the following operations using recursive implementations

(i) Append [3]
(ii) Delete [4]
(iii) Display [3]

(b) Write a function to append one linked list to another. [10]

QUESTION SEVEN

(a) Provide a detailed description of the Big-O notation. [6]
(b) Explain how it may be used with respect to an algorithm of your choice. [14]

END OF QUESTION PAPER

GOOD LUCK!