

NATIONAL UNIVERSITY OF SCIENCE AND TECHNOLOGY  
FACULTY OF APPLIED SCIENCE  
COMPUTER SCIENCE DEPARTMENT  
DECEMBER EXAMINATIONS 2000

SUBJECT: ADVANCED PROGRAMMING

CODE: SCS 4105

INSTRUCTION TO CANDIDATES

Answer any five [5] questions  
All questions carry equal marks [20 each]

Time: 3 hours

1. Use the operations *push* (which adds an element to a stack), *pop* (which deletes an element from a stack), *stacktop* (which returns the top element on a stack), and *empty* (which returns true if the stack is empty and false otherwise), to construct operations which do the following

- a) Set  $i$  to the second element from the top of the stack, leaving the stack without its top two elements.
- b) Set  $i$  to the second element from the top of the stack, leaving the stack unchanged.
- c) Given an integer  $n$ , set  $i$  to the  $n$ th element from the top of the stack, leaving the stack without its top  $n$  elements.
- d) Given an integer  $n$ , set  $i$  to the  $n$ th element from the top of the stack, leaving the stack unchanged.
- e) Set  $i$  to the bottom element of the stack, leaving the stack empty.
- f) Set  $i$  to the bottom element of the stack, leaving the stack unchanged. [Hint use another auxiliary stack]
- g) Set  $i$  to the third from bottom element of the stack.

[20]

2a) Let  $comm(n,k)$  represent the number of different committees of  $k$  people that can be formed, given  $n$  people from whom to choose. For example,  $comm(4,3)=4$ , since given four people A, B, C, and D there are four possible three-person committees: ABC, ABD, ACD, BCD. Write a recursive C program to compute  $comm(n,k)$  for  $n,k \geq 1$ .

[5]

b) Write a recursive C program to sort an array  $a$  as follows:

- i. Let  $k$  be the index of the middle element of the array
- ii. Sort the elements up to and including  $a[k]$
- iii. Sort the elements past  $a[k]$
- iv. Merge the two arrays into a single sorted array
- v. What is the name of this sorting procedure?

"LIBRARY USE ONLY"

[15]

3a) Show how to implement a queue of integers in C by using an array  $q[100]$ , where  $q[0]$  is used to indicate the front of the queue,  $q[1]$  is used to indicate its rear, and  $q[2]$  through  $q[99]$  are used to contain the queue elements. Show how to initialize such an array to represent the empty queue and write functions *remove*, *insert*, and *empty* representing the primitive operations on a queue for such an implementation.

[10]

b) Redo question 3a) using a struct to represent your queue.

[10]

4a) write algorithms to traverse non-empty binary tree in

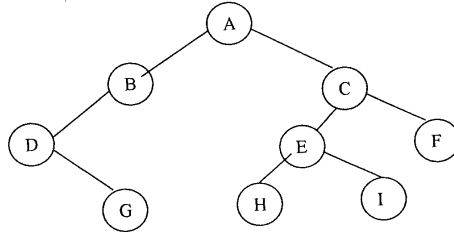
- i. preorder
- ii. inorder
- iii. postorder

[9]

b) Give the output of traversing the binary tree given below, by

- i. preorder
- ii. inorder
- iii. postorder

[6]



c) Write a C function that accepts a pointer to a node and returns *TRUE* if that node is the root of a valid binary tree and *FALSE* otherwise. [5]

5 a) Give an account of shortest path algorithm for graphs and implement this algorithm in C [20]

6. Briefly describe the following methods of selecting a free block of memory to be used when implementing dynamic memory management

- first-fit
- best-fit
- worst-fit

Give the algorithms for implementing any two of the above

[20]

7 a) Write functions to count the number of nodes in a binary tree, the number of leaves, number of right children, and the height of tree. [10]

b) Write a function to delete all leaves from a binary tree. [10]

**END OF QUESTION PAPER**