[3762]-204

S.E. (Computer Engineering)

(First Semester) EXAMINATION, 2010

DATA STRUCTURE AND ALGORITHMS

(Theory)

(2008 COURSE)

Time : Three Hours   **Maximum Marks : 100**

N.B. :— (i) Answer *three* questions from Section I and *three* questions from Section II.

(ii) Answers to the two Sections should be written in separate answer books.

(iii) Neat diagrams must be drawn wherever necessary.

(iv) Figures to the right indicate full marks.

(v) Assume suitable data, if necessary.

SECTION I

1. (a) Explain in brief, the different storage classes of variables in a 'C' function with examples.   [8]

    (b) Write a 'C' program to :

        (i) Read a matrix 'A' of real numbers of size $m \times n$.

        (ii) Calculate the mean of the elements of matrix 'A'.

(iii) Find the matrix 'B' containing deviation from mean of elements of A.

(iv) Display matrix 'B'

using functions with parameters. [8]

Or

2. (a) Write the output of the following code :

```c
#include<stdio.h>

#define M 5

void testoper(int);

int main( )

{

int i, m=4;

char a[ ]="abcd123";

for(i=1; i<m; ++i)

printf("%d\t%c\t%d\n", i, a[i], *(a + i));

testoper(M);

return 0;

}
```

```
void  testoper(int  x)

{

int  i = 10;

printf("%d%d%d",  i||x,  i&x,  i<<x);

}
```
                                                                                    [8]

(b) Write a 'C' program to read the 'n' records of books, with

     each book record containing the fields as book id, title, author

     and publisher. Store the records in a file. Also display the

     number of books written by a given author along with their

     book ids.   [8]

3. (a) Define the following, with examples :

     (i)   Linear data structure

     (ii)   Big O notation

     (iii) Space complexity

     (iv)   Data type.   [8]

(b) Explain the importance of data structure in the design of an

     algorithm.   [4]

(c) How do you determine the time complexity of an algorithm using frequency count ? [4]

Or

4. (a) Write an algorithm to find the multiplication of two matrices and determine the time and space complexity of your algorithm. [8]

(b) How do you implement the ADT list ? [4]

(c) Explain the characteristics of an algorithm with an example. [4]

5. (a) Write pseudo 'C' algorithm to find the addition of two sparse matrices. [6]

(b) How do you represent a polynomial using an array ? Write a 'C' function to read a polynomial containing 'n' terms. [6]

(c) Consider an integer array X[5][5]. Find the address of an element X[3][2], assuming the base address as 100, in row major and column major representation. [6]

Or

6. (*a*) Write pseudo 'C' algorithm to merge two sorted arrays of size 'm' and 'n' in to a third array. [6]

(*b*) What is spase matrix ? Write pseudo 'C' algorithm to find the simple transpose of a sparse matrix. Analyze the time complexity of the algorithm. [8]

(*c*) Derive the address calculation formula for a two-dimensional array 'X' in column major representation. [4]

SECTION II

7. (*a*) What is the need of searching and sorting ? [4]

(*b*) Write pseudo 'C' algorithm for quick sort and determine the time complexity. [6]

(*c*) Write the contents of the list and increment after each pass using shell sort for the following list of numbers : [6]

15 21 92 33 44 86 63 55

Or

8. (*a*) Explain in brief the index sequential search. [4]

(*b*) Differentiate between internal and external sorting. [4]

(c) Write pseudo 'C' algorithm for searching a given student name in an array of student names using binary search. Find the space and time complexity of your algorithm. [8]

9. (a) Explain the representation of a polynomial using linked list with an example. [4]

(b) How the linked organization is different from sequential organization ? Explain different types of linked lists. [6]

(c) Write 'C' functions to,

(i) Create a Doubly Linked List (DLL).

(ii) Add an element at the middle of the DLL. [6]

Or

10. (a) Give the applications of linked lists. [4]

(b) Write 'C' functions to perform the following operations on a singly linked list :

(i) Reverse the list

(ii) Delete a given element from the list. [6]

(c) Write 'C' functions to insert and delete the last element in a circular linked list. [6]

11. (a) What are the applications of a stack ? [4]

(b) Write pseudo 'C' algorithm to check the valid parenthesis in an arithmetic expression using stack. [6]

(c) How do you represent a multiple stack ? Give the algorithms for operations on a multiple stack. [8]

Or

12. (a) Convert the following expression into postfix expression and show the contents of stack : [4]

$$a - b * c + d / e$$

(b) Write short notes on :

(i) Josephus problem.

(ii) Simulation of recursion. [6]

(c) What is a double ended queue ? Write the algorithms for insert and display operations on it using linked list. [8]