Total No. of Questions—12]   [Total No. of Printed Pages—4

Seat
No.

[4857]-212

S.E. (Information Technology) (First Semester)
EXAMINATION, 2015
FUNDAMENTAL OF DATA STRUCTURES
(2008 PATTERN)

Time : Three Hours   Maximum Marks : 100

N.B. :—   (*i*)   Answers to the two sections should be written in separate answer-books.

(*ii*)   Answer any *three* questions from each Section.

(*iii*)   Neat diagrams must be drawn wherever necessary.

(*iv*)   Figures to the right indicate full marks.

(*v*)   Use of calculator is allowed.

(*vi*)   Assume suitable data, if necessary.

SECTION I

1.   (*a*)   Explain different bitwise operators in C.   [6]

(*b*)   Write a C program to print binary equivalent of a decimal number.   [6]

(*c*)   What do you mean by type-definition ? Explain with suitable example.   [4]

Or

2.   (*a*)   Write a C program to find out length of string without using library function.   [6]

(*b*)   Differentiate between macro and function with example. [6]

P.T.O.

(*c*)   What is output of the following C code ? Explain :   [4]

(*i*)   #define M(x) x*x

main()

{

        printf("%d", M(5+6));

}

(*ii*)   main()

{

        int x;

        x=5+6%–7;

        printf("%d", x);

}

3.   (*a*)   What is a pointer variable ? Explain declaration, initialization and accessing a pointer variable with an example.   [8]

(*b*)   Write a recursive function to calculate the factorial value of an integer entered through the keyboard.   [8]

(*c*)   Distinguish between (*a)[10] and *a[5].   [2]

Or

4.   (*a*)   Write a C program using pointers that compares two integer arrays to see whether they are identical. The function should return 1 if they are identical, 0 otherwise.   [10]

(*b*)   Explain the usage of command line arguments with an example.   [4]

(*c*)   Compare malloc and calloc function in C language.   [4]

5.   (*a*)   Write a C program for Fibonacci series and calculate its time complexity.   [6]

(*b*)   Define and explain time and space complexity of an algorithm. [6]

(*c*)   Compare linear and non-linear data structure.   [4]

Or

6. (*a*) Explain static and dynamic data structures. [6]

(*b*) What is frequency count ? Explain with an example. Find the frequency count for the given piece of code : [6]

add(a,b,c,m,n)
{
    for i:=1 to m do
        for j:=1 to n do
            c[i,j]:=a[i,j]+b[i,j];
}

(*c*) What is an abstract data type ? Explain with an example. [4]

SECTION II

7. (*a*) Write a C program for iterative and recursive binary search to find a number in a given list of sorted numbers. [10]

(*b*) Compare merge sort and quick sort. Comment on time and space complexity in best, average and worst cases for both. [8]

Or

8. (*a*) Write an algorithm for bubble sort. Consider the following set of numbers. Sort them using bubble sort and show all passes : [10]

20, –14, 8, 13, –12, 2, 6, 7.

(*b*) Consider the following numbers. Sort them using bucket sort and show all passes : [8]

54, 14, 88, 56, 25, 11, 45, 91, 30, 12, 4.

9. (*a*) Represent a sparse matrix using suitable data structure and write pseudo C code to subtract two sparse matrices. Analyze its time complexity. [10]

(*b*) Explain the concept of linear data structure with example. [6]

Or

10. (*a*) Write an algorithm for the fast transpose of sparse matrix. Compare simple and fast transpose. Comment on time and space complexity for both. [10]

    (*b*) Explain sequential memory organization with example. [6]

11. (*a*) Write data structure to represent the following generalized lists using linked list and represent them : [6]

    (*i*) (a, (b, c), d)

    (*ii*) (p, q, (r, s, (t, u), v)).

    (*b*) Write a C program to create doubly link list. [6]

    (*c*) Explain importance of header in a linked list. [4]

Or

12. (*a*) Write a C program to add two sorted circular linked list of polynomials to form a third sorted list. Write time complexity. [10]

    (*b*) Write node structure to represent GLL. Represent the following list using GLL : [6]

    (*i*) (a, (b, c), d)

    (*ii*) (a, b, c, (d, e)).