[3662]-212

S.E. (I.T.) (I Sem.) EXAMINATION, 2009

FUNDAMENTAL OF DATA STRUCTURE

(2008 COURSE)

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

N.B. :—   (*i*)   Answer any *three* questions from each Section.

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

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

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

SECTION I

1.   (*a*)   What is similarity between structure, union and enumeration ? [6]

(*b*)   What is Macro ? What are its advantages and disadvantages. [6]

(*c*)   Write C program to interchange two variables without using third variable.   [4]

Or

2.   (*a*)   Describe about storage allocation and scope global, extern, static, local and register variables.   [8]

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

P.T.O.

(c) Determine output of the following C statement : [4]

(i) main ( )

```
{      int x = 10, y = 5, p, q;

       p = x > 9;

       q = x > 3 & & y ! = 3 ;

       printf (" p = % d, q = % d ", p, q) ;
}
```

(ii) main ( )

```
{      int x ;
       x = − 3 + 4 − 7 * 8/5 % 10 ;
       printf (" x = % d ", x) ;

}
```

3. (a) Define Pointers ? How do we declare the pointers ? Give its advantages. [6]

(b) Differentiate between pass by reference and pass by value. [4]

(c) Explain the effect of the following statements : [6]

(i) int p, * p ;

(ii) int q, * p = & q ;

(iii) int (* p) + + ;

(iv) int p + + ;

(v) char * p ;

(vi) int a = * p + 5 ;

4. (*a*) What is difference between null pointer and null macro ? [6]

(*b*) What are pointers really good for ? [4]

(*c*) What is recursion ? Explain with example. [6]

5. (*a*) Define time and space complexity of an algorithm. [6]

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

(*c*) Define the following terms : [4]

    (*i*) Data object

    (*ii*) Data type.

(*d*) What is an abstract data type ? [2]

Or

6. (*a*) Define data structure. What are different types of data structure ? Explain. [8]

(*b*) Explain different asymptotic notation. [4]

(*c*) What do you mean by frequency count and its importance in the analysis of an algorithm. [6]

SECTION II

7. (*a*) Compare the selection sort and insertion sort with respective to : [8]

    (*i*) Time complexity

       (*ii*)   Passes

       (*iii*) Storage requirement

       (*iv*) Sort stability.

  (*b*) Sort the following data in ascending order using quick sort. Show all passes with pivot :

$$56,\ 12,\ 84,\ 56,\ 28,\ 0,\ -13,\ 47,\ 94,\ 31.$$ [8]

Or

8. (*a*) What are different applications of sorting methods.   [6]

    (*b*) Write pseudo code for binary search with recursion.   [6]

    (*c*) What is bubble sort ? Explain with example.   [4]

9. (*a*) What is sparse matrix ? What are its applications.   [6]

    (*b*) Represent the following polynomials using arrays :

       (*i*) $x^3 + 2xy + y^3 - y + x$

       (*ii*) $5x^2 - 10xy + y^2 - 20.$ [4]

    (*c*) Write a short note on Storage representation of order list. [6]

Or

10. (*a*) Write pseudo C algorithm for addition of two polynomials. [8]

    (*b*) Explain with example simple and fast transpose.   [8]

11. (*a*) Write C pseudo code to insert and delete an element from singly linked list.   [6]

(b) Compare linked list with arrays with reference to the following aspects :

(i) accessing any element randomly

(ii) insertion and deletion of an element

(iii) utilization of computer memory. [6]

(c) Discuss the applications of circular singly linked list in detail. [6]

Or

12. (a) Write comparison between sequential linked organization with linked organization. [6]

(b) Represent the following GLL :

(i) G = ((a, b), ((c, d), e))

(ii) (p, q, (r, s, (t, u, v), w) x, u). [6]

(c) Write C function that removes all duplicates elements from SLL. [6]