[3662]-264

S.E. (Computer) (I Sem.) EXAMINATION, 2009
DATA STRUCTURES AND ALGORITHMS
(2003 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*)   Neat diagrams must be drawn wherever necessary.

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

SECTION I

1. (*a*)   Explain the following terms with suitable examples :   [8]

(1)   Data and different types of data.

(2)   Data structures

(3)   Abstract data types

(4)   Linear and non-linear data structures.

(*b*)   What is an algorithm ? Explain the different algorithm design tools ?   [6]

(*c*)   Explain the concept of static and dynamic data structures with examples.   [4]

Or

2. (*a*)   Explain the phases of :

(*i*)   Analysis

(*ii*)   Design

(*iii*)   Coding

(*iv*)   Verification with respect to program development. [4]

(*b*)   Determine the frequency counts for all statements in the following :

```
for (i = 1; i ≤ n; i++)
  { for (j = 1; j ≤ i; j++)
      { for (k = 1; k ≤ j; k++)
          { x = x + 1;
          }
      }
  }
```
[6]

(*c*)   Draw a flow-chart for recursive binary search, also write its pseudo code. [8]

3. (*a*)   Give an array as ADT |Abstract data type]. [5]

(*b*)   Explain the concept of row major and column major in storage representation of arrays. Give 'C' program examples in support of the concepts. [5]

(*c*)   Write a pseudo 'C' code to find the fast transpose of a sparse matrix. [6]

Or

4. (*a*)   Write a pseudo 'C' algorithm to add two polynomials, also give suitable data structures for the formation of a polynomial. [8]

(*b*)   Write a pseudo 'C' code to find the simple transpose of a sparse matrix. [6]

(*c*)   What is the concept of multidimensional arrays ? Give example in support. [2]

5. (a) Give stack as an ADT. [6]

(b) (i) What are the different stack operations ? Explain with examples. [4]

(ii) Explain the concept of multiple stack. [2]

(c) What is recursion ? What are the advantages of recursive functions ? [4]

Or

6. (a) Explain how a stack is used for reversing a string. Give diagrammatic support for the explanation. [6]

(b) Convert the following infix expressions into postfix and prefix form using a stack : [10]

(i) $(A + B) * C - (D + E) \wedge ((F - G) + H)$

(ii) $A/(B + C) * D - G$.

SECTION II

7. (a) Write queue as an ADT and explain. [6]

(b) Explain the concept of double ended queue. [4]

(c) Write an algorithm that reverses the contents of a queue. [8]

Or

8. (a) Explain Josephus problem as an application of queues. [6]

(b) Explain the concept of priority queues. [4]

(c) Write an algorithm called 'stack to queue' that creates a queue from a stack. After the queue has been created the top of stack should be front of the queue and base of stack should be rear of queue. At the end of algorithm the stack should be empty. [8]

9. (a) Give the comparison between quick sort and selection sort, with respect to : [6]

    (i) Complexity

    (ii) Stability.

(b) What are the some of sorting considerations that must be addressed in any sorting technique ? [4]

(c) Give the merge sort algorithm with complete explanation. [6]

Or

10. (a) Explain the concept of sorting and stability in sorting techniques. [4]

(b) What are the differences between sequential and index sequential search ? [4]

(c) Sort the following numbers using shell sorting : [8]

    18 40 25 90 100 02 89.

11. (a) (i) What is the need for analysis of an algorithm ?

    (ii) Prove that the complexity of selection sort algorithm is $O(n^2)$. [8]

(b) Explain the concept of space and time complexity with suitable examples. [4]

(c) What is the algorithmic strategy in the 'Towers of Hanoi' problem ? [4]

Or

12. (a) Give the algorithm for non-recursive binary search and analyse it for best case. [6]

(b) Explain the concept of back-tracking in the 8 queens problems. [6]

(c) Give any *two* characteristics of 'Divide and Conquer' and give any *two* applications of the same. [4]