Reg. No. : | | | | | | | | | | | |

K 4211

B.E./B.Tech. DEGREE EXAMINATION, MAY/JUNE 2009.

Third Semester

(Regulation 2004)

Electronics and Communication Engineering

CS 1151 — DATA STRUCTURES

(Common to B.E/B.Tech. Second and Third Semester Computer Science and Engineering and Information Technology branches of Annual Pattern candidates admitted in 2006)

(Also common to B.E. (Part-Time) First Semester Computer Science and Engineering Regulation 2005)

Time : Three hours Maximum : 100 marks

Answer ALL questions.

PART A — ($10 \times 2 = 20$ marks)

1. Define methods of program design.

2. Define PROGRAM COMPLEXITY and state how it is computed.

3. State the advantages of ABSTRACT DATA TYPE.

4. State the properties of LIST abstract data type with a suitable example.

5. List the operations defined on BINARY TREES data type with an example.

6. List the operations defined on PRIORITY QUEUES with a suitable example.

7. Discuss the advantages of QUICK SORT and state how it is achieved.

8. Discuss the types of EXTERNAL SORTING.

9. Describe the DIJKSTRA's method briefly.

10. Define the term MINIMUM SPANNING TREE.

PART B — ($5 \times 16 = 80$ marks)

11. (a) (i) Develop an algorithm for computing internal marks of a single subject having two internal tests of 10 marks each and one assignment of 20 marks. The marks are to be rounded to nearest integer and converted to a maximum of 20 marks by appropriate scaling. Assuming your class contains 60 students and all the students have appeared for all tests and assignment, compute the complexity of your algorithm. (8)

(ii) Define the terms Program Verification and Efficiency. Describe methods available for the same. (8)

Or

(b) Develop a suitable algorithm using TOP-DOWN-DESIGN for computing pay-checks, given the employee details and working details. Employee details are name of employee and hourly payment along with daily allowances. Working details contains number of hours worked per week and number of days. (16)

12. (a) Develop an algorithm for permitted operations on STACK data type. (16)

Or

(b) Develop an algorithm for insertion and deletion of names of 35 characters each on a QUEUE data type. (16)

13. (a) Develop an algorithm for searching required value on a SEARCH TREE. (16)

Or

(b) Develop an algorithm for implementing traversals on BINARYTREE data type. (16)

14. (a) (i) Discuss the issues in sorting. (6)

(ii) Develop an algorithm for implementing INSERTION SORT. (10)

Or

(b) (i) Develop an algorithm for implementing SHELL SORT. (8)

(ii) Develop a method to compute the complexity and efficiency of SHELL SORT. (8)

15. (a) Develop an algorithm for TOPOLOGICAL SORT. (16)

Or

(b) Develop an algorithm for implementing SHORTEST PATH algorithm from the user supplied list of nodes and a matrix of node connectivity. (16)
