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

Question Paper Code : 21352

B.E./B.Tech. DEGREE EXAMINATION, MAY/JUNE 2013.

Third Semester

Electronics and Communication Engineering

EC 2202/EC 33/10144 EC 303/080290009 — DATA STRUCTURES AND OBJECT ORIENTED PROGRAMMING IN C++

(Regulation 2008/2010)

Time : Three hours Maximum : 100 marks

Answer ALL questions.

PART A — (10 × 2 = 20 marks)

1. Define class and objects.

2. Define inline functions and friend functions.

3. What is inheritance?

4. Give an example for exception handling.

5. What is an DEQUE?

6. Write any two advantages of binary heap.

7. Why is always a red node inserted into a red-black-tree?

8. Does the minimum spanning tree of a graph give the shortest distance between any two specific nodes? Justify.

9. List the four types of sorting techniques.

10. How data is sorted in a QUEUE structure?

PART B — (5 × 16 = 80 marks)

11. (a) Write short notes on the following :

(i) Comparison of conventional programming and OOPS (6)

(ii) Operator overloading (6)

(iii) Constructors and destructors. (4)

Or

(b) (i) Explain the control structures of C++ with suitable examples. (12)

(ii) Define function over-loading with a simple example. (4)

12. (a) (i) Explain protected data with private and public inheritance. (8)

(ii) Write a C++ program for to solve eight queens problem with friend functions. (8)

Or

(b) Write an example program for virtual functions and pure virtual functions with suitable algorithm. (16)

13. (a) (i) Explain the operations performed on QUEUE in detail. Write a C++ program to implement these QUEUE operations. (10)

(ii) Write a program for insertion of a node in a binary heap. (6)

Or

(b) (i) Write a program to implement STACK through linked list. (8)

(ii) Explain the function of open-addressing and chaining in collision resolution. (8)

14. (a) Write a program to accept keys from the user one at a time, build them into an AVL tree and write out the tree at each stage. Also write a function to delete a node from AVL tree. (16)

Or

(b) Define spanning tree and minimal spanning tree. Write Kruskal's algorithm for finding minimal spanning tree of any graph. (16)

15. (a) (i) Compare bubble-sort with insertion-sort with an example. (8)

(ii) Explain how divide and conquer technique can be applied for merge sort. (8)

Or

(b) Find the expected number of passes, comparisons and exchanges for shell sort when the number of elements is equal to 10. Compare this result with the actual number of operations when the given sequence is as follows : (16)

7, 1, 3, 4, 10, 9, 8, 6, 5, 2.