UNIT – I

PRINCIPLES OF OBJECT ORIENTED PROGRAMMING

PART-A

1. State the characteristics of procedure oriented programming.

2. What are the features of Object Oriented Programming?

3. Distinguish between Procedure Oriented Programming and Object Oriented Programming. Define Object

Oriented Programming (OOP).

4. List out the basic concepts of Object Oriented Programming.

5. Define Objects.

6. Define Class.

7. Define Encapsulation and Data Hiding.

8. Define Data Abstraction.

9. Define data members and member functions.

10. State Inheritance.

11. State Polymorphism.

12. List and define the two types of Polymorphism.

13. State Dynamic Binding.

14. Define Message Passing.

15. List out some ofthe benefits of OOP.

16. Define Object Based Programming language.

17. List out the applications of OOP.

18. Define C++.

19. What are the input and output operators used in C++?

20. What is the return type of main ()?

21. List out the four basic sections in a typical C++ program.

22. Define token. What are the tokens used in C++?

23. Define identifier. What are the rules to be followed for identifiers?

24. State the use of void in C++.

25. Define an enumeration data type.

26. Define constant pointer and pointer to a constant.

27. What are the two ways of creating symbolic constants?

28. Define reference variable. Give its syntax.

29. List out the new operators introduced in C++.

30. What is the use of scope resolution operator?

31. ist out the memory differencing operator.

32. Define the 2 memory management operators.

33. List out the advantages of new operator over malloc ().

34. Define manipulators. What are the manipulators used in C++?

35. Define integral widening conversion.

36. What are the control structures used in C++?

37. Define Function Prototyping.

38. What is call by reference?

39. What are inline functions?

40. List out the conditions where inline expansion doesn’t work.

41. Why do we use default arguments?

42. State the advantages of default arguments.

43. Define function overloading.

44. List out the limitations of function overloading.

45. State the difference between structures and class.

46. Define a class.

47. List the access modes used within a class.

48. How can we access the class members?

49. Where can we define member functions?

50. What are the characteristics of member functions?

51. How can an outside function be made inline?

52. What are the properties of a static data member?

53. What are the properties of a static member function?

54. How can objects be used as function arguments?

55. Define friend function?

56. List out the special characteristics of a friend function.

57. Define Constructor.

58. List some of the special characteristics of constructor.

59.Give the various types of constructors.

60.What are the ways in which a constructor can be called?

61. State dynamic initialization of objects.

62.Define Destructor.

63. Give the general form of an operator function.

64. List some of the rules for operator overloading.

65. What are the types of type conversions?

66. What are the conditions should a casting operator satisfy?

PART – B

1) Explain with the Basic Concepts of object oriented programming. (16)

2) (a) Explain the elements of object oriented programming. (8)

(b) What are the difference between reference variables and normal variables? (8)

3) Explain about call-by-reference and return by reference. (16)

4) (a) Describe the advantages of OOP. (8)

(b) What are the difference between pointers to constants and constant to pointers? (8)

5) (a) Describe the applications of OOP technology. (8)

(b) What is function overloading? Explain with an example program. (8)

6) Explain the merits and demerits of object oriented methodology . (16)

7) What is friend function? What is the use of using friend functions in c++? Explain with a program. (16)

8) What is polymorphism? Provide an example to explain it. (16)

9) Describe Abstract base class. Illustrate an example to explain it. (16)

10) What are the advantages of using default arguments? Explain with example program. (8)

11) Write a program to implement nested classes using c++. (16)

12) Write a programto demonstrate how a static data is accessed by a static member function. (16)

13) Write a program to get the student details and print the same using pointers to objects and pointers to members of a class. Create a class student. And use appropriate functions and data members. (16)

14) Explain copy constructor and destructor with suitable C++ coding. (8)

15) Explain about Implementation of simple ADTs. (16)

16) Explain about staticmember and this pointer with suitable code. (16)

17) What is a Bit field? Explain briefly. (8)

18) Explain about Data Handling and member function. (8)

19) What is a virtual destructor? Explain the use of it. (16)

20) Explain about Unary Operator and Binary Operator Overloadingwith program. (16)

21) List out the rules for overloading operators with example. (8)

22) Define a supplier class. Assume that the items supplied by any given supplier are different

and varying in number. Use dynamic memory allocation in the constructor function to

achieve the solution. (16)

23) Write a C++ Program to check whether the string is palindrome or not? (8)

24) Write a C++ program to define Binary and unary operator. (8)

UNIT – II

ADVANCED OBJECT ORIENTED PROGRAMMING

PART-A

1. What are the types of inheritance?

2. Give the syntax for inheritance.

3. Define single inheritance.

4. Define multi-le vel inheritance.

5. Define multip le inheritance.

6. Define Hierarc hic a l inheritance.

7. Define Hybrid inheritance.

8. What is a virtua l base class?

9. What is an abstract class?

10. What are the types of poly morphism?

11. Define ‘this’ pointer.

12. What is a virtua l function?

13. What is a pure virtual function?

14. How can a private me mber be ma de inheritable?

15. What is hierarchical inheritance?

16.What is multilevel inheritance?

17.Mention some of the Separators used in Java Programming?

18.What is boolean data type?

19.How dynamic initialization of variables is achieved in java?

20.What is meant by Widening conversion?

21. What is meant meant by Abstract base class?

22. Write short notes on virtual base class.

23. What are the iteration statements used in C++?

24. What is the difference between break & continue statements?

25. What are the uses of break statements?

26. Mention some of the restrictions while using static keyword?

27. Define data members and member functions.

28. List and define the two types of Polymorphism.

29.Define Message Passing.

30. Define C++.

31. What is the return type of main ()?

32. State the use of void in C++.

PART – B

1. Explain the various types of Inheritance with suitable example program.

2. What are the virtual functions? Explain their needs using a suitable example. What are the rules associated with virtual functions?

3. What are the different forms of inheritance supported in c++? Discuss on the visibility of base class members in privately and publicly inherited classes.

4. What are abstract classes? Give an example (with the program) to illustrate the use of abstract classes. (8)

5.Explain multiple catch statement with help of suitable C++ coding. (16)

6.Explain about Code Reuse with program. (8)

7.Discuss about Run Time Type Identifications. (8)

8.Write notes on Typing conversions and derived class with program. (16)

9. Explain about Exceptions Handlers and Standard Exceptions. (8)

10.Explain about Template and its types with example. (16)

11. Discuss about Streams and stream classes. (8)

12. Write notes on Formatted and Unformatted Console I/O Operations. (16)

13. Explain about File Pointers and Manipulations with example. (16)

14. Discuss about manipulators and file streams with Program. (16)

15.Write on Details about File modes and File I/O. (16)

16.Write notes on Formatted and Unformatted Console I/O Operations. (16)

17.Explain about File Pointers and their manipulations with example. (16)

18.Give the differences between Manipulators and ios Functions. (8)

19.How can we determine errors while dealing with files. (16)

20.Explain in detail aboutthefacilities availableforsubstring operations on thestring object? (16)

21.Explain in detail about Sorted Associative Containers. (16)

22. Discuss about different ways of defining namespaces. (8)

23. Define a student class. Inherit that into MCAStudent class and NonMCAStudent. MCA Students inherits into GLSSTudents and NonGLSStudents. A function howPracticalHours can only be applied to MCAStudents. We have a base class Student pointer to a GLSStudent object.Use dynamic_cast ti check that NonMCAStudents do not ShowPracticalHours. (16)

UNIT – III

DATA STRUCTURES & ALGORITHMS

PART-A

1. Write down the definition of data structure.

2. What is anAlgorithm?

3. What are the properties of an Algorithm?

4. Define Program.

5. What is Complexity analysis?

6. Explain the performance analysis of the algorithm?

7. Explain Space complexity?

8. Explain Time complexity?

9. List out the components that are used for space complexity?

10. Define Efficiency of an algorithm?

11. Define Worst case of an algorithm?

12. Define Best case of an algorithm?

13. Define average case an algorithm?

14. Define Divide and Conquer algorithm?

15. Mention some application of Divide and Conquer algorithm?

16. Define dynamic programming algorithm?

17. Mention application of dynamic programming algorithm?

18. State the various steps in algorithm?

19. Define algorithm paradigms space of an algorithm?

20. Mention the various spaces utilized by a program?

21. Define ADT (Abstract Data Type)?

22. Define linear data structure?

23. Define Non Linear data structure?

24. What are different types of Linked List?

27. Define a stack?

28. Define a queue?

29. What is single linked list?

30. Define HEAD pointer and NULL pointer?

31 What is meant by dummy header?

32. Define Circular linked list?

34. Mention applications of stack?

35. Define Infix, prefix and postfix notations?

36. What are the conditions that followed in the array implementation of queue?

PART-B

1. A)Explain Top down design in detail.

B) Design an algorithm that accepts positive integer and reverse the order of its digits (8)

2. Write the ADT for find routine in the linked list. (16)

3. Write an algorithm to

a) To count the numbers nodes in the linked list (4)

b) Reverse a singly linked list (4)

c) Concatenate tow single linked list (4)

d) Copy one linked list to another (4)

4. Given singly linked list whosefirstnodeis pointed to by the pointervariable C

formulate an algorithm to delete the first occurrence of X from the list and Insert

the elementX afterthe position Pin thelist. (16)

5. Explain the process of postfix expression evaluation with an example. (16)

6. Write the program to print the elements of single linked list. (16)

7. Explain with an example the creation of linked list, insertion and deletion of nodes and swapping of any two nodes. (16)

8. Usinglinked listformulate separateroutinesto create an empty stack and push an element on to the stack. (16)

9. Discussin detail the arrayimplementation of queue (16)

10. Explain the implementation stack using linked list. (16)

11. Explain prefix infix and postfix expressions with example. (16)

12. Explain the operations and the implementations of list ADT. (16)

13. Give a procedure to convert an infix expression a+b*c+(d*e+f)*g to postfix notation. (8)

14. Design and implement an algorithm to search a linear ordered linked list for a given alphabetic key or name. (16)

15. Define an efficient representation of two stacks in the given area on memory with n words and explain. (16)

16. What is a stack? Write down the procedure for implemention various back operations (8)

17. Explain the various application of stack? (16)

18. Given two sorted lists L1 and L2 write a procedure to impurt L1_ L2 using only the basic Operation (8)

19. Write a routine to insert an element in a linked list (16)

20. What is a queue? Write an algorithm to implement queue (16)

21. Explain the process of conversion from infix expression to postfix using stack (16)

22. Explain the process of postfix expression evaluation with Example (16)

23. Write the ADT operation to test whether current position is the last in the linked list (16)

24. Write the ADT operation for insertion and deletion routin in linked lists (16)

25. Write example describe how you will measure the efficiency of an algorithm (16)

26. Analyze the search algorithm with example (16)

27.Explain thevariousaspects ofproblemsolving in detail. Also describethe pros and cons of each (8)

28. Write the ADT operation for insertion and deletion routine in linked Queue (16)

29. Write a suitable C routine to remove and return the top element of stack using Array implementation? (16)

30. Formulate a routine to implement a stack using linked list and to pop and element Into the stack. (16)

31. Write a routine to implement a queue using arrays and to enqueue and element in to it. (16)

32. List out the ADT operation to enqueue and Dequeue and element in queue. (16)

33.List outthe ADT operations of circularlinked list? (16)

34. Define hash function? Write the routines to find and insert an element in separate chaining. (16)

35. Explain rehashing techniques to avoid collision. (16)

36. Explain extendible hashing to resolve collision. (16)

37. Define Heap? State the properties of heap. Explain the Maximum heap and minimum heap with an example? (16)

UNIT – IV

NONLINEAR DATA STRUCTURES

PART-A

1. Define tree?

2. Define non-linear data structure?

3. What is a Binary tree?

4. What are the applications of binary tree?

5. What is meant by traversing?

6.What is binary tree traversal?

7. What are the different types of traversing?

8. What are the two methods of binary tree implementation?

9. Define Graph?

10. Define adjacent nodes?

11.Name the different ways of representing a graph?

12. What are the two traversal strategies used in traversing a graph?

13. What is an acyclic graph?

14. Give some example of NP complete problems.

15. Define Height of tree.

16. Define Depth of tree.

17. Define Degree ofa node.

18. Define Degree ofa tree.

19. Define Terminal node or leaf?

20. Define Non-terminal node?

21. Define sibling?

22. Define binary tree?

23. Define expression tree?

24. Define Constructionofexpression trees

25. Define lazy deletion?

26. Define AVL

27. What are the various operation performed in the binary search tree?

28. General idea of hashing and what is the use of hashing function?

29. What is priority queue?

30. Applicationofpriorityqueues?

31. What are the main properties of a binary heap?

32. Define tree traversal and mention the type of traversals?

33. What is a graph?

34. What are Directed graphs?

35.Define path.

36.Define Cycle.

37. Define Acyclic graph.

38. What are the conditions for a graph to become a tree?

39. What is a minimum spanning tree?

40. Explain about Adjacency Matrix.

41. Explain about Adjacency linked list.

42.What isasinglesourceshortestpathproblem?

43. Explain about Unweighted shortest path.

44. Explain about Weighted shortest path

45. What are the methods to solve minimum spanning tree?

46.Explainbriefly aboutPrimalgorithm

47. Define a depth first spanning tree.

PART-B

1. Explain the operations of insertion of nodes into and deletion of nodes from, a binary search

tree with code. (16)

2. Give the analysis of insertion and deletion operations of nodes in binary

search tree. (8)

3. Write a algorithm to locate an element from binary search tree. (8)

4. Draw the binary search tree for the following input list 60,25,75,15,50,66,33,44.Trace

an algorithm to delete the nodes 25,75,44 from the tree. (16)

5.Explain the operations ofinsertion ofnodes into and deletion ofnodesfrom, a binarysearch

tree with code. (16)

6. Explain the basic terminologies in trees. (16)

7. How do you insert an element in a binary search tree? (8)

8. Discuss the advantages of linked list representation of trees over array representation. (8)

9. List out the ADT operations of Binary search tree. (8)

10. Explain BFS and DFS with example. (16)

11. Define AVL trees? Explain the LL, RR, RL, LR case with an example? (16)

12. Explain in detail insertion into AVL Trees. (16)

13. Explain in detail (i) Single rotation (ii) double rotation of anAVL tree. (16)

14. Write an Algorithm to insert a node in AVL tree. (8)

15. Write an Algorithm to delete a node from AVL tree. (8)

16. Write the ADT Routines for AVL tree operation. (8)

17. Explain wit example how u insertion and deletion of a node in AVL tree? Explain the all the possible cases. (8)

18. Explain the various representation of graph with example in detail? (16)

19. Define topological sort? Explain with an example? (16)

20. Explain Dijkstra's algorithm with an example? (16)

21. Explain Prim's algorithm with an example? (16)

22. Explain Krushal's algorithm with an example? (16)

23.Explain minimum Spanning tree . (16)

24. Design a single source shortest path algorithm and explain it with example. (16)

25. What are the Graph traversal method? Explain it with example. (16)

26. Give an algorithm to find minimum Spanning tree, Explain it with suitable example. (16)

27.Write an algorithm for finding minimum spanning tree and explain application, illustrate the algorithm with typical data of yours own example. (16)

28. Explain the algorithmfor depthfirst search and breathfirstsearchwith thefollowing graph (16)

UNIT – V

SORTING AND SEARCHING

PART-A

1. What is meant by sorting?

2. What are the two main classifications of sorting based on the source of data?

3. What is meant by external sorting?

4. What is meant by internal sorting?

5. What are the various factors to be considered in deciding a sorting algorithm?

6. What is the main idea in Bubble sort?

7. What is the basic idea of shell sort?

8. What is the purpose of quick sort?

9. What is the advantage of quick sort?

10. What is the average efficiency ofheap sort?

11. Define Algorithm.

12. What is complexity analysis?

13. What is performance analysis of an algorithm?

14. Define space complexity.

15. Define time complexity.

16. What does asymptotic notation mean?

17. Define best case of an algorithm.

18. What is divide and conquer technique?

19. What is dynamic programming?

20. Define Greedy method.

21. Who invented shell sort? Define it?

22. Write the function in c for shell sort?

23. What is maxheap?

24. What are the two stages for heap sort?

25. What is divide and conquer strategy?

26. Mention some methods for choosing the pivot element in quicksort?

27. What are the three cases that arise during the left to right scan in quicksort?

28. What is the need of external sorting?

29. Define two way merge?

30. Define multi way merge?

31. Define polyphase merge?

32. What is replacement selection?

33. What is sorting?

34. What is mergesort?

35. What are the properties involved in heapsort?

36. Define articulation points.

37. Give some example of NP complete problems.

PART-B

1. Sort the sequence 3, 1, 4,7,5,9,2,6,5 using Insertion sort. (16)

2. Explain the operation and implementation of Insertion sort and shell sort. (16)

3. Explain the operation and implementation of merge sort. (16)

4. Explain the operation and implementation of external sorting. (16)

5. (a) Write quick sort algorithm and explain. (8)

(b) Trace the quick sort algorithm for the following list of numbers.

90,77,60,99,55,88,66 (8)

6. Write down the merge sort algorithm and give its worst case, best case and

average case analysis. (16)

7. Show how heap sort processes the input 142,543,123,65,453,879,572,434,111,242 ,811,102. (16)

8.Write ADT operation for Heap sort. Using the above algorithm sort the following

35,45,25,11,6,85,17,35 (16)

9. Explain the Quick sort algorithm with example (16)

10. Explain external sorting. Give an example (16)

11. State and explain to perform heap sort with example (16)

12. Analyse the time complexity of quick sort algorithm. (16)

13. Explain anyone external sortingwith an example (16)

14. Briefly explain about quick sort algorithm (16)

15. Trace the quick sort algorithm for the following numbers 90,77,60,99,55,88,66 (16)

16. Explain the stages of heap sort with example (16)

17. Explain the quick sort algorithm with example (16)

19. Explain the Heap sort algorithm with example (16)

20. Briefly explain about Greedy algorithm with example (16)

21. Briefly explain about Dynamic programming (16)

22. Explain the divide and conquer technique with the help of merge sort (16)