



AKSHEYAA COLLEGE OF ENGINEERING
Puludivakkam, Maduranthagam Taluk, Kancheepuram - 603314



Department of Electronics and Communication
Engineering

EC6301 / OOPS & Data Structures (Regulation-2013)

II YEAR / III Semester

UNIT-WISE EXPECTED UNIVERSITY EXAMINATION QUESTIONS

UNIT I – DATA ABSTRACTION & OVERLOADING

Part-A

1. What is a reference variable? **Apr/May 2015**
2. What is a friend function? **Apr/May 2015**
3. Why there is need for operator overloading? **Apr/May 2015**
4. Write a C++ code to swap value of two variable using reference variables in function. **Nov/Dec 2014 regulation-2013**
5. Write a C++ code to display “pen object instantiated” and “pen object destroyed” when class for pen constructor and destructor are called. **Nov/Dec 2014 regulation-2013**
6. What is data encapsulation? Give example. **Nov/Dec 2013**
7. When do we declare member of a class static? **Nov/Dec 2013**
8. How to create symbolic constants in C++? **May/June 2014**
9. Define destructor with syntax. **May/June 2014**
10. Define class and objects. **May/June 2013**, Define object. **Nov/Dec 2012**
11. Define inline and friend function. **May/June 2013**
12. Define constructor. **Nov/Dec 2014**, What is constructor? **Nov/Dec 2012**
13. What is operator over loading? Enlist the operators that cannot be overloaded. **Nov/Dec 2014**, Mention the operators that cannot be overloaded. **May/June 2012**, List out the operators which cannot be overloaded. **Nov/Dec 2012**
14. How to initialize a pointer? **Nov/Dec 2014**
15. Highlight the advantage of static data member and static member function in C++. **May/June 2012**
16. Define pure virtual functions and its usage. **May/June 2012**
17. What is the significance of contour structures? **Nov/Dec 2011**

Part-B

1. Describe the **major components of object oriented programming** with illustration. **Apr/May 2015**, Explain the characteristics of object oriented programming in detail. **May/June 2012**
2. What is the purpose of **constructor and destructor**? Explain with suitable example the different types of constructor in C++? **Apr/May 2015, Nov/Dec 2014, Nov/Dec 2011 (or)** Explain with syntax multiple constructors in a class. **May/June 2014 (or)** what is constructor with default arguments? **May/June 2014** Explain with example the type's constructors in C++. **Nov/Dec 2013**, Illustrate the working of constructors and destructors with an example. **May/June 2012**
3. Write a member function and **friend function** to subtract two complex numbers in C++. **Nov/Dec 2014 regulation-2013**, Write a C++ program to swap two numbers using friend function. **Nov/Dec 2014**
4. Write a member function to perform matrix addition, simple addition and string concatenation by overloading + operator. **Nov/Dec 2014 regulation-2013**
5. Write a C++ program that contains a class string and overloads the following operators on strings. **Nov/Dec 2013**
 - + to concatenate two strings
 - To delete a substring from the given string
 - == to check for the equivalence of both strings
6. Explain call by reference and return by reference with syntax. **May/June 2014**
7. Write short notes on the following **May/June 2013**
 - [i] Comparison of conventional programming and OOPS
 - [ii] **Operator overloading**
 - [iii] Constructor and destructor.
8. Explain **control structures of C++** with suitable example. **May/June 2013, Nov/Dec 2014, Nov/Dec 2011**
9. Define function overloading with a simple example. **May/June 2013**
10. Explain the structure of C++ program with an example. **Nov/Dec 2014**
11. Write a program to evaluate the equation, $A=B*C$ using classes and objects where A, B and C are objects of the same class **May/June 2012**
12. Write a menu driven program to accept 2 integers and an operator (+, -, *, %, /) and to perform the operation and print the result. **Nov/Dec 2012**
13. Specify the class complex to represent complex numbers. Overload +, -, / and * operators when working on the objects of this class. **Nov/Dec 2012**
14. Define friend function. What is polymorphism? Explain multiple inheritances. **Nov/Dec 2012**
15. Explain the functions of C++ with suitable example. **Nov/Dec 2011**

UNIT II – INHERITANCE & POLYMORPHISM

Part-A

1. What is overriding? **Apr/May 2015**
2. Write a C++ code to display as area of square or rectangle using function overriding. **Nov/Dec 2014**
3. Write a sample code to show the usage of this pointer in C++. **Nov/Dec 2014**
4. What is the need to declare base classes as virtual? **Nov/Dec 2013**
5. What is the use of virtual functions in C++? **Nov/Dec 2013**
6. List out various forms of inheritance. **May/June 2014**
7. What is inheritance? **May/June 2013**, What does multiple inheritance mean? **Nov/Dec 2012**
8. What is run time polymorphism? **Nov/Dec 2014**
9. What is polymorphism? **Nov/Dec 2011**

Part-B

1. What is inheritance? Discuss in detail about the various **types of inheritance** in C++ with suitable examples. **Apr/May 2015 (or)** what is inheritance? Explain with example the different types of inheritance in C++? **Nov/Dec 2013**, Write a C++ program to illustrate the concept of hierarchical inheritance. **Nov/Dec 2014**, Define inheritance .Mention its types. Write a C++ program to illustrate the concept of multiple inheritances. **May/June 2012**
2. What is **virtual function**? Explain with an example how late binding is achieved using virtual function. **Apr/May 2015**
3. Write a C++ code to construct classes of a person with name and age as public properties, account details as private properties and percentage of mark as protected property. Construct a class with sports details of person. Construct a class to rank person based on the equal weight age to academic and sports details. Use inheritance concept. **Nov/Dec 2014 regulation-2013**, Create an base class named 'shape' with two members base and height , a member function for initialization and a virtual function to compute area () . Derive two specific classes' triangle and rectangle which override the function area () . Use classes in the main function and display the area of a triangle and a rectangle using virtual functions. **May/June 2012**
4. Explain **class object to base and base to class object conversion** using C++ with suitable example. **Nov/Dec 2014 regulation-2013**
5. Derive inheritance for insurance policies. **May/June 2014**
6. Give the structure form of scope rules for public, private and protected access to super class and sub class member and objects. **May/June 2014**
7. Explain **polymorphism** with an example **May/June 2014**
8. Explain protected data with private and public inheritance **May/June 2013**
9. Write a C++ program for to solve eight queens problem with friend functions. **May/June 2013**
10. Write an example program for virtual functions and **pure virtual functions** with suitable algorithm. **May/June 2013**, Write a C++ program to implement virtual function. **Nov/Dec 2014**, Explain about the virtual functions. **Nov/Dec 2011**

UNIT III – LINEAR DATA STRUCTURES

Part-A

1. What is ADT? **Apr/May 2015**, Define ADT. **Nov/Dec 2012**
2. Write short notes on queue. **Apr/May 2015**
3. Evaluate the value of expression $ab + c*d$ using stack. **Nov/Dec 2014 regulation-2013**
4. What is a priority queue? **Nov/Dec 2013**
5. Define recursion and specify the data structures used to perform recursion. **Nov/Dec 2013**
6. What is an DEQUE? **May/June 2013**
7. How data is stored in a Queue structure? **May/June 2013, Nov/Dec 2014, Nov/Dec 2011**
8. What are the limitations of linear queues? How are they overcome using circular queues? **May/June 2012**
9. What is meant by underflow and overflow condition in a stack? **May/June 2012**
10. What are the basics data structures available in C++? **Nov/Dec 2012**

Part-B

1. Write a set of routines for implementing two **stacks** within a single array. **Apr/May 2015**
2. Write a set of routines for implementing queue using linked list. **Apr/May 2015**, With a simple program explain various operations of linked list. **Nov/Dec 2014**
3. Write a C++ code to sum up all odd numbers in a single link list. **Nov/Dec 2014 regulation-2013**
4. Write a C++ code to perform addition of two **polynomials** using link list form of queue. **Nov/Dec 2014 regulation-2013**
5. Explain the process of inserting and deleting an element in a circular queue with an example. **Nov/Dec 2013**
6. What is **queue**? Explain by comparing with stack. **May/June 2014**
7. Write pseudo code for insertion and deletion from queue. **May/June 2014**
8. What is the need of linked list? **May/June 2014**
9. Give syntax to insert X in **linked list**. **May/June 2014**
10. Explain the operation performed on queue in detail. Write a C++ program to implement these queue operations. **May/June 2013**
11. Write a program to implement Stack through linked list. **May/June 2013**, Briefly discuss about stack and its operations **Nov/Dec 2014**, Write pseudo code for the following: [a] Split a stack into two. The first contains the bottom half elements and the second contains the remaining elements [b] Combine two stacks by placing all elements of the second stack on top of those in the first stack. **May/June 2012**, Explain the operations performed by stack in detail. Write a C++ program to implement these operations. **Nov/Dec 2011**
12. Write an algorithm to perform each of the following operations.
 - [i] Reverse a list, so that the last element comes first and so on.
 - [ii] Return the sum of integers in a list.
 - [iii] Delete every third element from a list. **May/June 2012**
13. Write algorithms for insertion and deletion in linked stack. **Nov/Dec 2012**
14. Design an algorithm to split a circular list into two circular lists. **Nov/Dec 2011**
15. Formulate an algorithm to create an ordered linear list with no duplicate terms. The input is a polynomial with three variables that is unordered and has repeated terms. **Nov/Dec 201**

UNIT IV – NON LINEAR DATA STRUCTURES

Part-A

1. What is a tree? **Apr/May 2015**
2. How a graph is represented? **Apr/May 2015**
3. Find the maximum number of nodes in complete binary tree if d is the depth. **Nov/Dec 2014 regulation-2013**
4. Write short notes on connected components. **Nov/Dec 2014 regulation-2013**
5. Give the representation of network of cities (Chennai, Delhi, Kolkata and Mumbai) as weighted graph. **Nov/Dec 2014 regulation-2013**
6. How to perform union operation? **Nov/Dec 2014 regulation-2013**
7. Write the template for depth first search. **May/June 2014**
8. What is binary tree? Give example. **Nov/Dec 2013**
9. Differentiate between tree and graphs. **Nov/Dec 2014, Nov/Dec 2011**
10. When the tree is called complete binary tree? **Nov/Dec 2012**

Part-B

1. Discuss the **different methods traversing a binary tree** with algorithm. **Apr/May 2015**, Explain binary tree with syntax. **May/June 2014**
2. Illustrate the **depth first search algorithm** with a graph and explain. **Apr/May 2015**, What is graph? Explain the depth first search tree. **Nov/Dec 2012**
3. Explain DFS and **BFS** with suitable example. **Nov/Dec 2014 regulation-2013**
4. Write C++ code for the implementations of different types of tree traversal. State few tree applications. **Nov/Dec 2014 regulation-2013**
5. Explain **binary search tree** with its operations. [or] Create binary search tree for the following alphabets. Start from an empty BST R, F, G, B, Z, U, P, L, K, L delete keys B, U and L one after the other and show the trees at each stage **May/June 2012**

UNIT V – SORTING AND SEARCHING

Part-A

1. What is meant by sorting? **Apr/May 2015**
2. What is time complexity? **Apr/May 2015**
3. What is the time complexity of quick sort and binary search? **Nov/Dec 2014 regulation-2013**
4. State why quick sort is more efficient than merge sort. **Nov/Dec 2013**
5. What is K-way merge? **May/June 2014**
6. List the four types of sorting techniques. **May/June 2013**
7. Sort the numbers 34, 12, 25, 14 using merge sorting technique. **Nov/Dec 2014**
8. Differentiate between stable and unstable sorts. **May/June 2012**
9. What is the worst case and best case no. of comparison in a linear search? **Nov/Dec 2012**

Part-B

1. Discuss the **quick sort** algorithm and apply the same for the following numbers 90,77,60,99,55,88,66 **Apr/May 2015**
2. Explain in detail about **linear search algorithm** with an example. **Apr/May 2015**
3. Write C++ code to implement quick sort with suitable example. Write C++ code to implement linear search with suitable example. **Nov/Dec 2014 regulation-2013**
4. Write C++ code to implement **merge sort** with suitable example. Write C++ code to implement binary search with suitable example. **Nov/Dec 2014 regulation-2013, Nov/Dec 2012**
5. Sort the following values using quick sort and estimate its time and space complexity: 65, 70, 75,80,85,60,55,50,45 Illustrate each step of the sorting process. **Nov/Dec 2013**, Sort the following using quick sorting 15, 25, 70, 07, 11, 65, 81, 57 **Nov/Dec 2014**
6. For which sorting divides and conquers technique is used. Write its algorithm with explanation to sort 10 vales. **May/June 2014**
7. Give short notes of: [i] merge sort with suitable example. [ii] Quick sort with suitable example. **May/June 2014**
8. Explain how divide and conquer technique can be applied for merge sort. **May/June 2013, May/June 2012**
9. 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: 7, 1, 3, 4, 10, 9,8,6,5,2] **May/June 2013**
10. Explain **insertion sort** with its time complexity. **Nov/Dec 2014**
11. With example the **binary search technique**. **Nov/Dec 2012**

“Concentrate more on bolded headings”