Sri vidya college of Engineering and Technology

Department of Computer science and Engineering

Sub. Name: Object oriented programming and data structures   Sub. Code: EC6301

Sem/Class: III/II-ECE   Staff name: M.Kavipriya

Two Mark Questions

UNIT-1

1. List out some of the benefits of OOP.

    Reusability: In OOP's programs functions and modules that are written by a user can be reused by other users without any modification.

    Inheritance: Through this we can eliminate redundant code and extend the use of existing classes.

    Data Hiding: The programmer can hide the data and functions in a class from other classes. It helps the programmer to build the secure programs.

    Message Passing: The technique of message communication between objects makes the interface with external systems easier.

2. Differentiate: Object oriented programming(c++) and Structures oriented programming(c).

S. No.	C	C++
1	C is a structural or procedural programming language.	C++ is an object oriented programming language.
2	Emphasis is on procedure or steps to solve any problem.	Emphasis is on objects rather than procedure.
3	Functions are the fundamental building blocks.	Objects are the fundamental building blocks.
4	In C, the data is not secured.	Data is hidden and can't be accessed by external functions.
5	C follows top down approach.	C++ follows bottom up approach
6	C uses scanf() and printf() function for standard input and output.	C++ uses cin>> and cout<< for standard input and output.

3. List down the basic concepts of OOP.
    1. Objects
    2. Classes
    3. Data abstraction and encapsulation
    4. Inheritance
    5. Polymorphism
    6. Dynamic binding
    7. Message passing

4. What are the applications of OOP
    a. Real time systems
    b. Simulation and modelling
    c. Object oriented data bases
    d. Hypertext, hypermedia and expertext
    e. AI and expert systems
    f. Neural networks and parallel programming
    g. Decision support and office automation systems
    h. CIM\CAM\CAD systems

5. Define Class.

    A class encloses both the data and functions that operate on the data into single unit. The Variables and functions in a class are called data member and member functions respectively.

    class classname

        {

            Private:Variable declaration;

            Function declaration;

            Public:Variable declaration;

            Function declaration;

        };

6. Define Object.

    An *object* is an instantiation of a class.

    Syntax for creating objects

        Class class_name object_name_1,............Object_name_n;

    Eg: class student s1,s2,s3;

7. Define encapsulation.

The wrapping up of data & functions into a single unit is known as data encapsulation. Here the data is not accessible outside the world.

8. What are data members & member function?

The attributes of a class is called data members.

Eg: int a,b; //a,b are data members

The function which operate on the data is called member function.

Eg: void getdata();//member function

9. List the access modes used within a class.

Public - The members declared as Public are accessible from outside the Class through an object of the class.

Protected - The members declared as Protected are accessible from outside the class but only in a class derived from it.

Private - These members are only accessible from within the class. No outside Access is allowed.

10. Differentiate: class and structure.

Class	Structure
A class encloses both the data and functions that operate on the data into single unit.	Structure is collection of dissimilar data items
All members of a class are private by default	All members of a Structure are public by default

class class_name
 	
{
 	
  Private:
 	
   Variable declaration;
 	
    Function declaration;
 	
Public:
 	
Variable declaration;
 	
Function declaration;
 	
};
	struct structure_name
	
	{
	
	Variable declaration1;
	
	Variable declaration1;
	
	.
	
	.
	
	Variable declaration_n;
	
	};

11. What are the parts of class specification?

The class specification has two parts

1. Class declaration – To describe the type and scope of its members

2. Class function definition – To describe how the class functions are implemented.

12. Write the syntax of class declaration

class classname
{
    Private:Variable declaration;
    Function declaration;
    Public:Variable declaration;
    Function declaration;
    };

13. Where will you define a member function?

Member functions can be defined in two places.

1. outside the class definition.

2. inside the class definition.

14. Give the syntax for member function definition outside the class.

Return_type class_name:: function_name (argument list)
{
function body;
}

15. List the characteristics of member function.

1. Member function can access the private data of class.

2. A member function can call another member function directly without using the dot operator.

16. Define nesting of member function.

A member function can be called by using its name inside another member function of the same class. This is known as nesting of member functions.

17. List the properties of static members.

A data member of a class can be qualified as static properties.

1. It is always initialized to zero when the first object of its class is created.

2. It is visible only within the class.

18. Write the properties of static member function.

1. A static function can have access to only other static members declared in the same class.

2. A static member function can be called using the class name (Instead of objects) as follows: class name:: function name;

19. What is constructor? (Nov/Dec 2012)

A constructor is a member function, which takes the same name as the class name, used to initialize objects of its class. The constructor is automatically invoked when an object is created.

Eg: Class integer

{

Public:

Integer(); //constructor;

};

20. What are the characteristics of constructor?

1. Constructors should be declared in public section.

2. They are involved automatically when the objects are created.

3. They do not have return types.

4. They cannot be inherited.

21. What is a default constructor? (Nov/Dec 2011)

It is a type of constructor which does not hold any parameters. If it has, those parameters should have default values.

22. Define parameterized constructor.

Arguments can be passed to the constructor function when the objects are created. The constructors that can take arguments are called as parameterized constructor.

23. What is an implicit constructor?

C++ compiler has an implicit constructor which creates objects even though it was not defined in the class.

24. What is the use of copy constructor?

Copy constructor is used to declare and initialize an object from another object.

E.g. Class name object2 (object1);

25. What do you mean by dynamic construction?

Allocation of memory to objects at the time of their contraction Is known as dynamic contraction of objects.

26. What is the use of destructor?

It is used to destroy the objects that have been created by a constructor. It releases the memory space for future use.

27. What are the characteristics of destructor?

1. A destructor is a member function whose name is the same as the class name but it is preceded by a tilde.

2. It neither takes any argument nor returns any value.

3. It will be invoked implicitly by the compiler to clean up the storage

28. Define operator overloading.

Operator overloading can be defined as an ability to define a new meaning for an existing operator.

29. Mention the operators which cannot be overloaded (April/may2012)

➢ Scope resolution operator (::)

➢ Sizeof operator (sizeof)

➢ Conditional operator (?:)

➢ Class member access operator (.)

30. Define function overloading.

Function overloading is a concept in which one can use many functions having the same name but can pass different number of parameters or different types of parameters.

Example:

```
Int sum(int a,int b);

Int sum(int a,int b,int c);

Int sum(int a,int b,int c,int d);
```

31. What are friend functions? Write the syntax.

A function that has access to the private member of the class but is not itself a member of the class is called friend function.

Syntax: friend data_type function_name();

32. What is Dynamic memory allocation?

Allocation of memory space for any data structure during the course of the program execution is called as Dynamic memory allocation.

33. How memory management is performed dynamically in C++?

Two operators are available for dynamic memory management.

(i) new – for dynamic memory allocation

(ii) delete – for dynamic memory deallocation

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

➤ **new** operator automatically computes the size of the data object. You need not use the operator **sizeof**.

➤ **new** operator automatically returns the correct pointer type, so that there is no need to use a type cast.

➤ It is possible to initialize the object while creating the memory space.

➤ Like any other operator, **new** and **delete** can be overloaded.

16 Mark Questions

1. Illustrate the working of constructors and destructors with an example.
2. Explain the control structures of C++ with suitable examples.
3. Illustrate the use of copy constructors and function overloading with C++ programming.
4. Specify a class called complex to represent complex numbers. Overload +, - and * operators when working on the object of this class.
5. Define friend function specify its importance. Explain with suitable example.
6. What is parameterized constructor? Explain with example.
7. What is operator overloading? Overload the numerical operators + and / or complex numbers "addition and division" respectively.
8. Write short notes on container classes and proxy classes.
9. Write short notes on Dynamic Memory Allocation and Static Class Members.
10. Briefly explain about oops concepts.
11. . Write a C-++ program that contains a class String and overloads the following operators on Strings.
    ➤ +to concatenate two strings
    ➤ -to delete a substring from the given String
    ➤ ==to check for the equivalence of both string   (Nov/dec 2013).

12. Write a Menu driven program for accept two integers and an operator to perform the operation (+,-,*,/,%) and print the result(16)   (Nov/dec 2012)
13. Specify the class called complex to represent the complex numbers overload (+,-,*) operators When working on the objects of this class?(16)   (Nov/dec 2012)
14. Explain the merits and demerits of Object oriented paradigm?(6)   (Nov/dec 2011)
15. Write a c++ Program to define overloaded constructors and to perform string initialization and String copy (10)(Nov/dec 2011)
16. Define a class 'string'. Use overload '= =' operator to compare two strings. (8) (Nov 2009)