# Question Bank

**for**

# Object Oriented Programming using C++ (16/17MCA22)

**Module 1:**

1. Distinguish Procedural oriented programming and object oriented programming.
2. Explain the concepts of object oriented programming.
3. Write a note on *bool* data type in C++ with suitable example.
4. Discuss string types in C++.
5. What do you mean by *inline* functions? Discuss with example.
6. Define class. Given syntax and example.
7. Differentiate class and structures in C++.
8. Define constructor and destructor. Illustrate working of constructors and destructors with suitable program.
9. Discuss order of execution of constructors and destructors.
10. Explain different types of constructors with suitable examples.
11. How do you pass objects to functions? Explain with example.
12. How can you return an object type from a function? Explain with a code snippet.
13. Explain static member functions and static data members in a class using suitable examples.
14. What is the significance of friend functions in C++? Explain with example.
15. What are friend classes? Discuss with a program.
16. Write a note on
    a. Scope resolution operator
    b. Nested classes
17. Programming examples on the topics like classes, objects, inline functions etc.

**Module 2:**

1. Discuss reference data types in C++ with suitable example.
2. Briefly discuss dynamic memory allocation in C++ with suitable code snippets.
3. How do you create an array of objects? How are they initialized? Discuss with example.
4. Discuss the significance of *this* pointer.
5. How do you create a pointer to an object and invoke class members? Explain with example.
6. Explain the concept of function overloading in C++ with suitable example.
7. Discuss default arguments with a code snippet.
8. Explain copy constructor with suitable program.
9. Write a note on
    a. Array of objects
    b. Ambiguity in function overloading (Function overloading resolutions)
10. Programming examples on the topics like reference variables, function overloading, default arguments etc.