Object Oriented Programming

1. What are object oriented concepts? What is difference between object-based, object-oriented and fully object-oriented language?
2. What are advantages of Object Oriented Programming? What is data security?
3. What is class and object? Give real-life example.
4. What are characteristics of object? Explain them.
5. What is the need of getter and setter functions in class?
6. What is abstraction and encapsulation. Give real-life example.
7. What is polymorphism? What are its types? Explain them with examples.
8. What is method overloading? Which are the rules of method overloading? Why return type is not considered in method overloading?
9. What are different types of hierarchy? When to use which one?
10. What is the difference between method overloading and method overriding?
11. What is object slicing? Explain object slicing in context of up-casting?
12. What is down-casting and when it is required? Explain with code.
13. What do you know about association, composition and aggregation? Explain with the help example.
14. What are different types of inheritance? Explain with the help of example. What are problems with multiple inheritance?
15. What is difference between interface, abstract class and non-abstract class? Which one to use where?
16. Which are the different types of design pattern? Explain singleton design pattern.

C++ Programming

1. How C++ manage its memory? Explain new & delete operators for variables, objects and arrays.
2. Is it possible to delete storage acquired by a local variable in static memory using delete keyword in C++?
3. What is the use of destructors? Write a legal example of a destructor.
4. What are the OOP concepts present in C++/Java? Explain with one example each.
5. How to create an abstract class in C++? How to write a pure virtual function? Can we write a body of pure virtual function?
6. What is getter and setter? Why do we use that? Write down a small code example.
7. Create a spiral matrix. (You can find this problem in Leetcode https://leetcode.com/problems/spiral-matrix/)
8. What is "this" pointer? Is it available for static, virtual, const and friend functions?
9. What is the need to write a user defined destructor? When should it be declared as "virtual"?
10. How does virtual function affect the size of an object? How are they executed at runtime?
11. What is the diamond problem? How to solve it?
12. What is shallow copy and deep copy? How is it implemented in C++? Explain with examples.
13. What is a smart pointer? Which are smart pointers in C++?
14. What is STL? Explain different components in STL with examples?