Контрольные вопросы:

1. Что определяет класс? Чем обличается класс от объекта?

2. Можно ли объявлять массив объектов? А массив классов?

3. Разрешается ли объявлять указатель на объект? А указатель на класс?

4. Допускается ли передавать объекты в качестве параметров, и какими

способами? А возвращать как результат?

5. Как называется использование объекта одного класса в качестве поля

другого класса?

6. Является ли структура классом? Чем класс отличается от структуры?

7. Объясните принцип инкапсуляции.

8. Что такое композиция?

9. Для чего используются ключевые слова public и private?

10.Можно ли использовать ключевые слова public и private в структуре?

11.Существуют ли ограничения на использование public и private в

классе? А в структуре?

12.Обязательно ли делать поля класса приватными?

13.Что такое метод? Как вызывается метод?

14.Может ли метод быть приватный?

15.Как определить метод непосредственно внутри класса? А вне класса? Класс

в C++ определяет тип данных, который содержит члены, такие как функции (методы) и переменные (данные). Объект является экземпляром класса, то есть реализацией этого класса. Класс описывает состояние и поведение, тогда как объект представляет конкретный экземпляр этого класса.

1. В C++ можно объявлять массив объектов, а также массив указателей на объекты. Однако нельзя объявлять массивы переменных типа класса.
2. Да, можно объявлять указатели на объекты класса, а также указатели на классы.
3. Объекты можно передавать в функции в качестве параметров по значению, по указателю или по ссылке. Они также могут возвращаться из функций такими же способами.
4. Использование объекта одного класса в качестве поля другого класса называется композицией.
5. В C++ структура является особым типом класса. Основное различие между классом и структурой заключается в доступе к членам по умолчанию (public для структуры и private для класса).
6. Принцип инкапсуляции в C++ заключается в объединении данных и методов, которые работают с этими данными, в классе, и скрытии данных от прямого доступа из других частей программы. Это позволяет обеспечить безопасность и упрощает использование класса.
7. Композиция - это принцип объектно-ориентированного программирования, который позволяет включать один объект в качестве части другого объекта.
8. Ключевые слова public и private используются для определения доступа к членам класса. Public члены доступны из любого места программы, в то время как private члены доступны только внутри класса.
9. В структуре в C++ также можно использовать ключевые слова public и private для определения доступа к её членам.
10. В классе ограничения на использование public и private связаны с доступом к членам класса извне. В структуре также существуют ограничения, но они определяются по аналогичным принципам.
11. Нет, делать поля класса приватными не обязательно, но это рекомендуется для поддержания инкапсуляции.
12. Метод в C++ - это функция, определенная внутри класса. Он может быть вызван с использованием оператора точки и экземпляра класса, например: obj.method();
13. Да, метод может быть приватным, что означает, что он доступен только внутри класса.
14. Метод можно определить непосредственно внутри класса, добавив его определение в теле класса. Также его можно определить вне класса, используя оператор разрешения области (::) для указания принадлежности к конкретному классу.