**1. Difference between overloading a function and re-defining a function**

* **Overloading a function**: This refers to creating multiple functions with the same name but different parameter lists (either in type, number, or order) within the same scope. It is a compile-time feature.
* **Re-defining a function**: This happens when a derived class provides its own version of a function that exists in the base class, overriding the base class's implementation. It is related to inheritance and can be dynamic (runtime polymorphism) if the function is virtual.

**2. Difference between private members and protected members**

* **Private members**: Accessible only within the class they are defined in. They are not accessible by derived classes.
* **Protected members**: Accessible within the class they are defined in and by derived classes. However, they are not accessible outside the class hierarchy.

**3. Difference between member access specification and class access specification**

* **Member access specification**: Defines the accessibility of individual class members (private, protected, or public).
* **Class access specification**: Defines how the base class members are treated when inherited by a derived class (e.g., public, protected, or private inheritance).

**4. What is a base class? What is a derived class?**

* **Base class**: A class that is extended or inherited by another class. It serves as a foundation for creating more specialized classes.
* **Derived class**: A class that inherits the properties and methods of a base class, potentially adding or overriding members.

**5. What are friend functions and where are they implemented?**

* **Friend functions**: Non-member functions that have access to the private and protected members of a class. They are declared in the class using the friend keyword but implemented outside the class.

**6. Which is called first, the base class constructor or the derived class constructor?**

* **The base class constructor** is called first.

**7. Which is called first, the base class destructor or the derived class destructor?**

* **The derived class destructor** is called first.

**8. When the inheritance is private, the private methods in the base class are \_\_\_\_\_\_\_\_\_\_ in the derived class (in C++).**

* **a. Inaccessible**

**9. Can we pass parameters to the base class constructor through the derived class or derived class constructor?**

* **YES**

**10. C++ Inheritance relationship is?**

* **c. Is a**