**Frequently Asked C++ Interview Question**

**What are the difference between C and C++?**

1. C++ is a subset of C. As Most of the program written in C can also be executed in C++;
2. C follows procedure oriented Programming while C++ follows both object oriented as well as procedure oriented programming.
3. As C++ supports object oriented programming, it also provide some functions which are not present in C such as function overloading, inheritance, friend function and so on.
4. C++ supports exception handling while in C we still do it using if-else statement.
5. C++ supports references, we can create reference to any variable but in c we don’t have references.
6. In C we use printf() and scanf(), while in C++ we use cin and cout object to take input from standard input stream and putting output on standard output stream.

**What are the difference between C++ and Java?**

1. Java is platform-Independent while C++ is platform dependent.
2. Java is mainly used for application development while C++ is mainly used for system programming.
   1. Java used for developing web-based, enterprise and mobile application.
3. C++ supports Multiple Inheritance while java doesn’t support multiple inheritance but in java it is achieved using Interfaces.
4. C++ supports pointers while Java don’t support pointers, it uses pointers internally but one can’t write program on pointers in java.
5. C++ uses compiler only on the other hand java use both compiler as well as interpreter which make java platform independent.
6. C++ supports call by reference while java don’t support call by reference.
7. C++ don’t have built in support for threads, while Java has built-in support for threads.
8. Java is a pure object oriented programming language while C++ follows both object oriented programming as well as procedure oriented programming.

**What are the similarities and difference between references and pointers?**

Similarities: -

1. Pointers and references both are used to change the value of local variable of a function in another function.
2. Both are used to gain efficiency, as we don’t need to copy whole object when we pass it into the function, we can simply pass a pointer or a reference of that object.

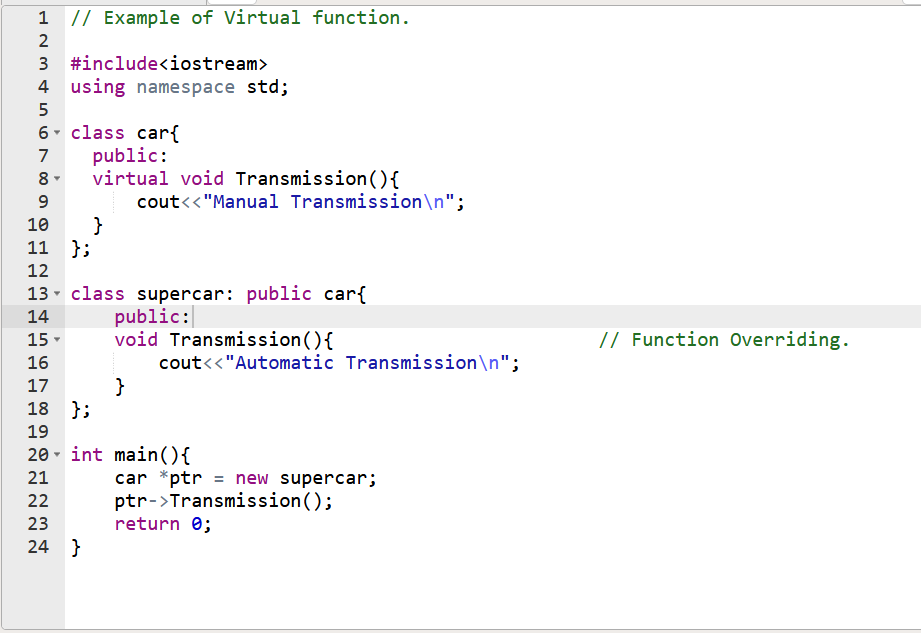
Differences: -

1. References must be initialized at the time of declaration while this is not the case with pointers.
2. We can create Null pointers while we can create a reference to NULL.
3. References are safe because of the restriction that it must be initialized, wild references like wild pointers don’t exist.
4. References are easy to use as we don’t need to dereference operator to access the value. They can be simply used like a normal variable.

**What is virtual functions? Example also.**

1. Virtual functions are used with inheritance, they are called according to the type of object pointed or referred, not according to the type of pointer or reference.
2. Virtual functions are not resolved at compile time. They are resolved at runtime.
3. Virtual keyword is used to make a function virtual.
4. Virtual function only comes in use when there is function overloading is taking placing. Which means it requires inheritance of base class to derived class, a function must be overridden in derived class and a pointer of base class type must be pointing to object of derived class.

Example: -



Output: - Automatic Transmission

**What is This pointer?**

1. This pointer is hidden argument to all non-static function calls and is available as a local variable within the body of all non-static functions.
2. ‘This’ pointer is a constant pointer that holds the memory address of the current object.
3. ‘This’ pointer is not available in static member functions as static member functions can be called without any object (with class name).

**What is the difference between a shallow and a deep copy of a class?**

Youtube.

**Difference between function overloading and function overriding?**

* Overloading provides multiple definitions of function by changing signature i.e changing number of parameters, change datatype of parameters, return type doesn’t play any role. On the Other hand, It is the redefinition of base class function in its derived class with same signature i.e return type and parameters.
* **Inheritance:** Overriding of functions occurs when one class is inherited from another class. Overloading can occur without inheritance.
* **Function Signature:** Overloaded functions must differ in function signature ie either number of parameters or type of parameters should differ. In overriding, function signatures must be same.
* **Scope of functions:** Overridden functions are in different scopes; whereas overloaded functions are in same scope.
* **Behavior of functions:** Overriding is needed when derived class function has to do some added or different job than the base class function. Overloading is used to have same name functions which behave differently depending upon parameters passed to them.

**What is template function?**

**What is Diamond Problem? How can we get around it?**

**Why are destructors important?**

**Is it possible to have a recursive inline function?**

**When you should use virtual inheritance?**

**Is there a difference between class and struct?**

**Explain the volatile and mutable keywords?**

**What is storage class?**

**What is class in C++? Give an example of class? What are the characteristics of class members in C++?**

**What is C++ objects? Example code snippet?**

**What is namespace std and what is consists of?**

**Explain what are Access specifiers in C++ class? What are the types?**

**What is the C-style character string? How it is different from C++ string class?**

**What is Polymorphism in C++?**

**What is data abstraction in C++?**

**What is C++ Exceptional Handling?**

**What is Data Encapsulation in C++?**

**What are the types of Member Functions?**

**What is Multi-Threading in C++?**

**What is upcasting in C++?**

**What is pre-processor in C++?**

**What is Copy Constructor and what is it used for?**

**What are Structures and Unions? What is the difference between Structures and Unions?**

**What do you mean by Enumerated Data Type?**

**What are Preprocessor Directives in C?**

**What is Pointers?**

**What do you mean by Static Variables?**

**Difference between global and Static Variables?**

**What do you mean by Volatile Variable?**

**Define Macro? What is advantages and disadvantages of using a macro?**

**What is void Pointer?**

**What is Dangling Pointer?**

**What is NULL Pointer?**

**What is Friend Class?**

**What is Overloaded operator?**

**What is Recursion? What is the advantages and disadvantages of using recursion? Example for which recursion is well suited?**

**What is Smart Pointers?**

**What is type casting?**

**What is lambda function? A simple program that use lambda function.**

**What is the difference between call by values and call by reference?**

**What is class template? Advantages of using class templates and Write a simple code that use class templates.**