COEN-244
Tutorial #5

February 16th, 2023

OOP: Inheritance

In C++, it is possible to inherit attributes and methods from one class to another. This is called **Inheritance**.

Two categories:

- **derived class** (child)
- **base class** (parent)

Why **Inheritance**?

+ It offers code reusability
+ It improves code readability
+ It saves time and effort

Modes of Inheritance in C++

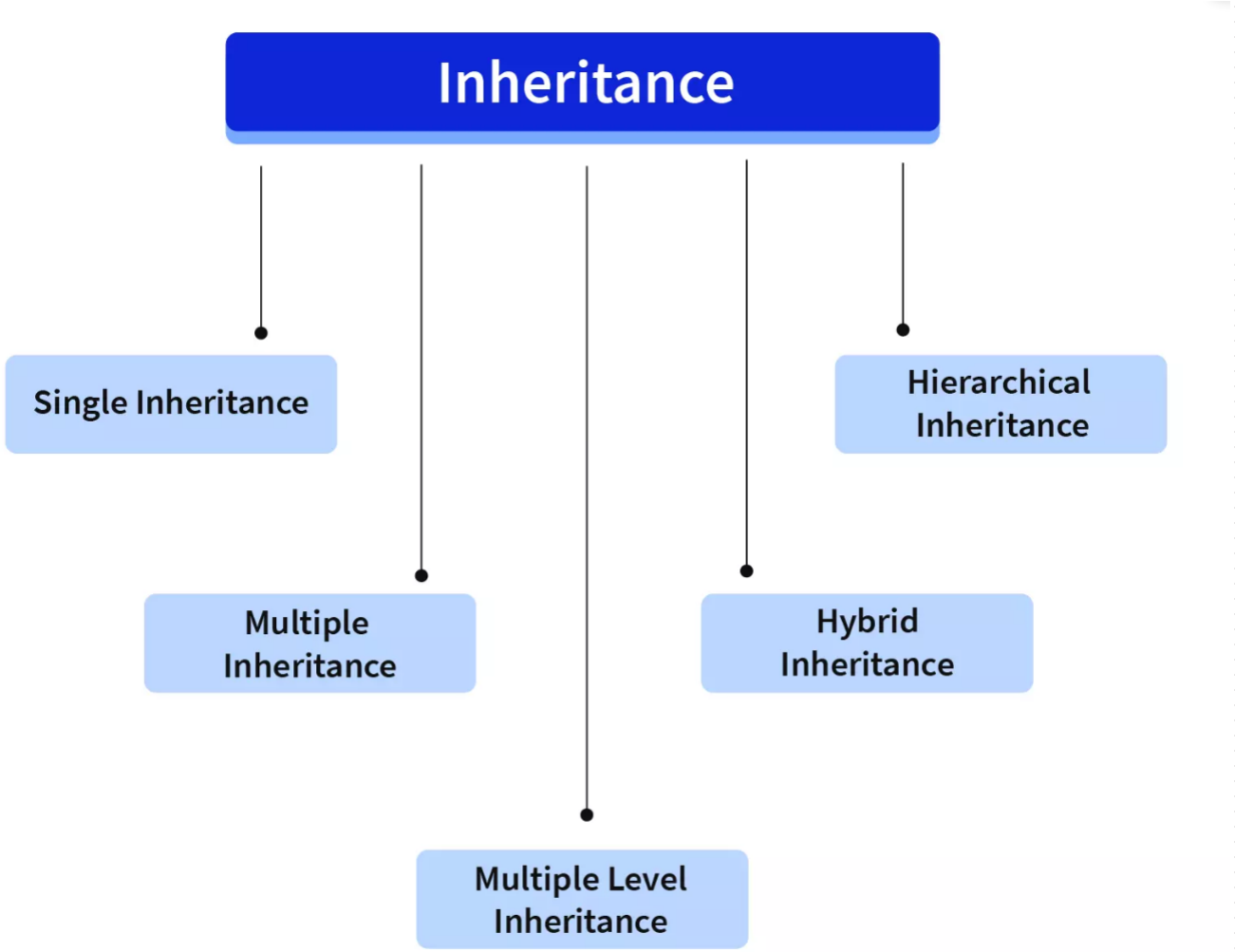
- **Public Mode:**
  - `Public` members are inherited as `public`
  - `Protected` members are inherited as `protected`
  - `Private` members are not inherited
- **Protected Mode:**
  - `Public & protected` members are inherited as `protected`
  - `Private` members are not inherited
- **Private Mode:**
  - `Public & protected` members are inherited as `private`
  - `Private` members are not inherited

Comparing Modes

Base class member access specifier	Type of Inheritence		
	Public	**Protected**	**Private**
Public	Public	Protected	Private
Protected	Protected	Protected	Private
Private	Not accessible(Hidden)	Not accessible(Hidden)	Not accessible(Hidden)

Source: https://www.scaler.com/topics/cpp/inheritance-in-cpp/

Types of Inheritance



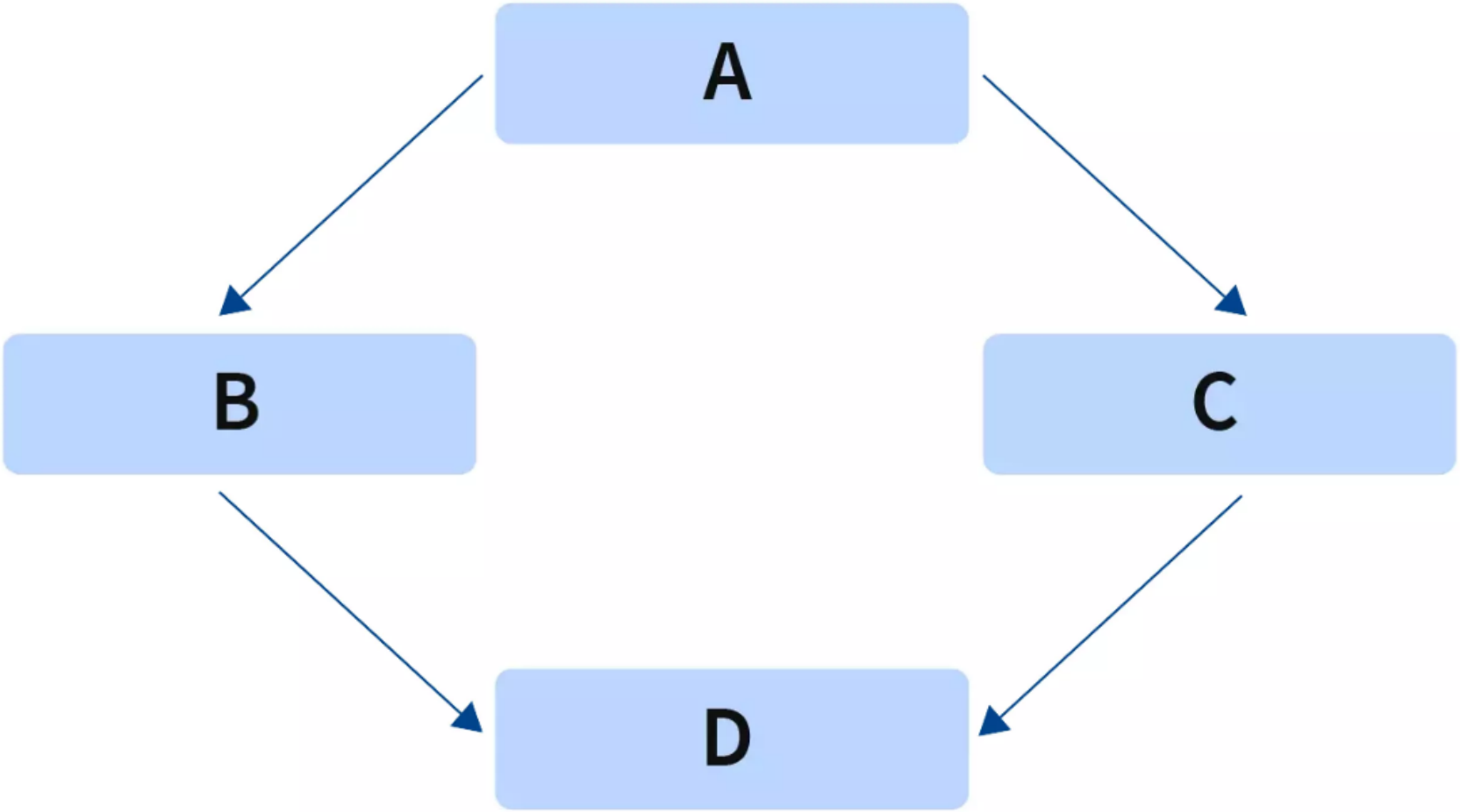
Source: https://www.scaler.com/topics/cpp/inheritance-in-cpp/

Problems in Inheritance

Problems in Inheritance mainly rise from **multiple inheritance.**

- **Diamond problem**
- **Members with the same identity**
- **Yo-Yo problem:** Jumping between base and derived class
- **Inheritance of Unnecessary Members**

DIAMOND PROBLEM



Source: https://www.scaler.com/topics/cpp/inheritance-in-cpp/

POLYMORPHISM

Polymorphism simply means more than one form.

- The same entity (function or operator) behaves differently in different scenarios

Different ways to do POLYMORPHISM:

- Static and dynamic binding + virtual functions
- Function overloading
- Function overriding
- Virtual destructors
- Pure virtual functions, Pure abstract classes

THANK YOU