

C++ in depth

Overriding, hiding and overloading



Saurabh Shukla (MySirG)

Agenda

- ① Function Overriding
- ② Function Hiding
- ③ Function Overloading

```
class A
```

```
{
```

```
public:
```

```
void f1() { ... }
```

B obj;

obj.f1(); //array

obj.f2(); //B f2()

obj.f1(5); //B f1(int)

```
}
```

```
class B : public A
```

```
{
```

```
public:
```

```
void f1(int x) { ... }
```

```
void f2() { ... }
```

```
}
```

Function
Hiding

Function Overloading

functions with the same name but differ in arguments and reside in single scope

is known as overloaded functions, and this phenomenon is known as function overloading or function polymorphism.

All overloaded versions of a function must be in the same class.

Function Hiding

When child class has a function with the same name but differ in arguments as compare to the version of function of parent class , we say , function in the child class hides the version of parent class . This concept is called function hiding.

One version must be in base class and the other version should be in a derived class of any of its descendant class.

```
class A
{
public:
    void f1() { ... }
};

class B : public A
{
public:
    void f1() { ... }
};

B obj;
obj.f1(); // B
```

function
Overriding

Function Overriding

Same prototype function, one version reside in base class and other version resides in derived (or any descendant class) class, then version defined in derived class overrides the version defined in base class.

This concept is known as function overriding.

Why overriding?

```
class Car
{
    gear
public:
void shiftGear() {...}
```

```
SportsCar obj;
obj.shiftGear();
```

```
};

class SportsCar : public Car
{
public:
void shiftGear() {...}

};
```

Whenever you want to modify a class
then do Inheritance

Whenever you want to modify a method
of base class with respect to derived
class, then do Overriding