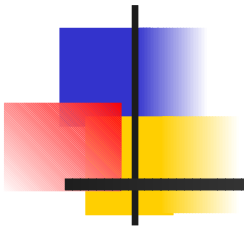
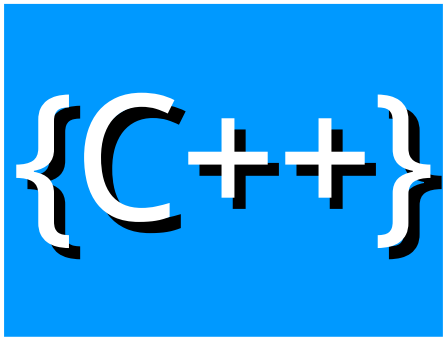


Polymorphism

Week 9



Yang-Cheng Chang
Yuan-Ze University
yczhang@saturn.yzu.edu.tw

Accessing Members of Base and Derived Classes

```
class B {
public:
   void m();
   void n();
   ...
} // class B

class D: public B {
public
   void m();
   void p();
   ...
} // class D
```

Class `D` *redefines* method `m()`

```
B B_obj;
D D_obj;
B *B_ptr = &B_obj;
D *D_ptr = &D_obj;
```

The following are legal:

```
B_obj.m() //B's m()
B_obj.n()

D_obj.m() //D's m()
D_obj.n() //B's n()
D_obj.p()

B_ptr->m() //B's m()
B_ptr->n()

D_ptr->m() //D's m()
D_ptr->n() //B's n()
D_ptr->p()
```

Accessing Members of Base and Derived Classes

```
class B {
public:
   void m();
   void n();
   ...
} // class B

class D: public B {
public
   void m();
   void p();
   ...
} // class D
```

Class `D` *redefines* method `m()`

```
B B_obj;
D D_obj;
B *B_ptr = &B_obj;
D *D_ptr = &D_obj;
```

- The following are legal:

```
B_ptr = D_ptr;
```

- The following are *not* legal:

```
D_ptr = B_ptr;
B_ptr->p();
```

Even if `B_ptr` *is known to point to an object of class* `D`

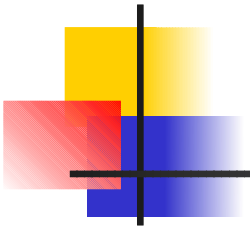
Virtual functions

```
class B {
public:
   virtual void m();
   void n();
   ...
} // class B

class D: public B {
public
   void m();
   void p();
   ...
} // class D
```

Class `D` *redefines* method `m()`

The following are legal:–

```
D D_obj;
B *B_ptr = &D_obj;

B_ptr = D_ptr;
B_ptr->m() //D's m()
B_ptr->n() //B's n()
```



Polymorphism

- The ability to declare functions/methods as **`virtual`** is one of the central elements of polymorphism in *C++*

- *Polymorphism:*– from the Greek “having multiple forms”
  - In programming languages, the ability to assign a different meaning or usage to something in different contexts

Assignment 9

- 請參考主程式的流程與物件宣告，完成三個類別的實作
  - 一個基礎類別 pet
    - 需要有建構子，用來初始化成員變數 name ，設定寵物名字
    - 一個成員變數 name， 用於儲存寵物的名字
    - 兩個成員函式 Name()，makeSound()
      Name() 傳回寵物的名字
      makeSound() 傳回寵物的叫聲，用字串代表
  - 兩個衍生類別 cat，dog ，繼承於基礎類別 pet
    - cat，dog 兩個類別的 makeSound() 必須不同，cat 的 makeSound() 傳回 miau，dog 的 makeSound() 傳回 won

主程式

```
#include <iostream>
#include "pet.h"

using namespace std;

void examinePet(pet* p){
    cout << "My name is " << p->Name() << " and I make " << p->makeSound() << endl;
}

int main(){
    pet* marry = new cat("marry");
    pet* tom = new dog("tom");
    examinePet(marry); // output: My name is marry and I make miau
    examinePet(tom); // output: My name is tom and I make won
}
```