继承与多态程序实例
– 多重继承

【例3-9】定义一个动物类Animal，成员变量包括：int型的m_nWeight（重量）。成员函数包括：SetWeight(int nWeight)、GetWeight()。

由Animal类派生出马类Horse，添加成员函数：

Run() { cout<<"I can run"<<endl; }

由Animal类派生出鸟类Bird，添加成员函数：

Fly() { cout<<"I can fly"<<endl; }

由Horse类和Bird类共同派生出天马类Pegasus。要求：

执行以下主函数，得出正确结果：

```
int main()
{
    Pegasus p;
    p.run();
    p.fly();
    p.SetWeight(5);
    cout<<p.GetWeight()<<endl;
    return 0;
}
```

```
// Animal.h
#ifndef _ANIMAL_H
#define _ANIMAL_H
#include <string>
#include <iostream>
using namespace std;
class Animal
{
public:
    Animal()
    { cout<<"Animal类的构造函数被调用！"<<endl; }
    ~Animal()
    { cout<<"Animal类的析构函数被调用！"<<endl; }
    void SetWeight(int nWeight)
    { m_nWeight = nWeight; }
    int GetWeight()
    { return m_nWeight; }
private:
    int m_nWeight;
};
#endif        // end of _ANIMAL_H
```

```
// Horse.h
#ifndef _HORSE_H
#define _HORSE_H

#include "Animal.h"
#include <iostream>
using namespace std;

class Horse : virtual public Animal
{
public:
    Horse() { cout<<"Horse类的构造函数被调用！"<<endl; }
    ~Horse() { cout<<"Horse类的析构函数被调用！" <<endl; }
    void run() { cout<<"I can run"<<endl; }
};

#endif        // end of _HORSE_H
```

```
// Bird.h
#ifndef _BIRD_H
#define _BIRD_H

#include "Animal.h"
#include <iostream>
using namespace std;

class Bird : virtual public Animal
{
public:
    Bird() { cout<<"Bird类的构造函数被调用！"<<endl; }
    ~Bird() { cout<<"Bird类的析构函数被调用！"<<endl; }
    void fly() { cout<<"I can fly"<<endl; }
};

#endif        // end of _BIRD_H
```

```
// Pegasus.h
#ifndef _PEGASUS_H
#define _PEGASUS_H
#include "Horse.h"
#include "Bird.h"
class Pegasus : public Horse, public Bird
{
public:
    Pegasus()
    {
        cout<<"Pegasus类的构造函数被调用！"<<endl;
    }
    ~Pegasus()
    {
        cout<<"Pegasus类的析构函数被调用！"<<endl;
    }
};
#endif        // end of _PEGASUS_H
```

```cpp
// main.cpp
#include "Pegasus.h"
int main()
{
    Pegasus p;
    p.run();
    p.fly();
    p.SetWeight(5);
    cout<<p.GetWeight()<<endl;
    return 0;
}
```

运行结果：

```
Animal类的构造函数被调用！
Horse类的构造函数被调用！
Bird类的构造函数被调用！
Pegasus类的构造函数被调用！
I can run
I can fly
5
Pegasus类的析构函数被调用！
Bird类的析构函数被调用！
Horse类的析构函数被调用！
Animal类的析构函数被调用！
```