```
#include<iostream>

using namespace std;

//赋值运算符重载

class Person{
public:
    Person(int age){
        this->m_Age=new int(age);
    }

    ~Person(){
        if(m_Age!=NULL){
            delete m_Age;
            m_Age=NULL;
        }
    }

    //重载 赋值运算符
    Person& operator=(Person& p){
        //编译器提供的是浅拷贝

        //应先判断是否有属性在堆区,如果有先释放干净,然后再深拷贝
        if(m_Age!=NULL){
            delete m_Age;
            m_Age=NULL;
        }

        //深拷贝
        this->m_Age=new int(*p.m_Age);

        return *this;
    }

    int* m_Age;


};

void test01(){
    Person p1=Person(18);
    Person p2=Person(20);
    Person p3=Person(30);

    p2=p1=p3;
```

```
    cout<<"p1的年龄为"<<*p1.m_Age<<endl;
    cout<<"p2的年龄为"<<*p2.m_Age<<endl;
    cout<<"p3的年龄为"<<*p3.m_Age<<endl;
}


int main(){
    test01();
}
```