```
#include<iostream>

using namespace std;

class Base1{
public:
    int m_A;
protected:
    int m_B;
private:  //三种都无法访问
    int m_C;
};

class Son1:public Base1{
public:
    void func(){
        m_A=10;  //父类中的公共权限成员 到子类中依然是公共权限
        m_B=10;  //父类中的保护权限成员 到子类中依然是保护权限
    }
};

class Son2:protected Base1{
public:
    void func(){
        m_A=10;  //保护权限
        m_B=10;  //保护权限
    }
};

class Son3:private Base1{
public:
    void func(){
        m_A=10; //私有权限
        m_B=10; //私有权限
    }
};


ostream& operator<<(ostream& cout,Base1& b){
    cout<<b.m_A<<"  ";
    return cout;
}

int main(){
    Son1 b1=Son1();
    b1.func();
    cout<<b1<<endl;
}
```