

```

// W12_2.cpp : Defines the entry point for the console application.
//          Hierarchcznosc funkcji wirtualnych przy dziedziczeni
//          B
//          !
//          D1
//          !
//          D2
//
// test 1. klasy pochodne D1, D2 maja definicje funkcji wirtualnej oper()
// test 2. klasa D2 nie ma definicji funkcji wirtualnej oper() - wywoływane D1::oper()
// test 3. klasy D2, D1 nie maja definicji funkcji wirtualnej oper() - B::oper(), B::oper()
// test 4. klasa D1 nie ma definicji funkcji wirtualnej oper() - B::oper(), D2::oper()

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

class B
{
protected:
    int i;
public:
    B(int ia) { i = ia; }
    int get_i() { return i; }
    virtual int oper() { cout << "B: " << endl; return 2*i; }
};

class D1 : public B
{
public:
    D1(int ia1): B(ia1) {}
    //jesli w klasie pochodnej funkcja wirtualna nie pozostala
    //przeslonieta, bedzie wywoływana realizacje klasy bazowej B
    //zakomentuj nastepna linie kodu i zobacz, ktora funkcje bedzie wywoływana
    virtual int oper() { cout << "D1: " << endl; return i*i*i; }
};

class D2 : public D1
{
public:
    D2(int ia2) : D1(ia2) {}

```

```

        //jesli w klasie pochodnej funkcja wirtualna nie pozostala
        //przeslonieta, bedzie wywoływana realizacje klasy bazowej B
        //zakomentuj nastepna linie kodu i zobacz, ktora funkcje bedzie wywoływana
        int oper() { cout << "D2: " << endl; return i*i*i*i; }
};

int _tmain(int argc, _TCHAR* argv[])
{
    B ob_b(1);
    D1 ob_d1(2);
    D2 ob_d2(3);

    B *ptr = &ob_b;
    cout << ptr->oper() << endl;

    ptr = &ob_d1;
    cout << ptr->oper() << endl;

    //jesli w klasie pochodnej definicje oper() pozostala opuszczona,
    //bedzie wywoływana metoda oper() klasy bazowej. Jesli pozostala zdefiniowana -
    //bedzie wywoływana realizacje oper() klasy pochodnej
    ptr = &ob_d2;
    cout << ptr->oper() << endl;

    //a to zwykle wywoływanie metody oper() klasy pochodnej.
    //jesli w klasie pochodnej definicje oper() pozostala opuszczona,
    //bedzie wywoływana metoda oper()klasy bazowej. Jesli pozostala zdefiniowana -
    //bedzie wywoływana realizacje oper() klasy pochodnej
    cout << ob_d2.oper() << endl;

    system("pause");
    return 0;
}

```