```
// W9_1.cpp : Defines the entry point for the console application.
//          Wielodziedziczenie      B
//                                *
//                                D1
//                                              *
//                                D2


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

//declaration of class base
class base
{
      int ib;
public:
      int jb;
      base(int ii, int jj) { ib = ii; jb = jj; cout << " konstruktor base\n"; }
      ~base() { cout << " destruktor base\n"; }
      int get_ib() { return ib; }
      void show() { cout << "base : ib = " << ib << " jb = " << jb << endl; }
};

//declaration of class derived_1
class derived_1 : public base
{
      int id;
public:
      int jd;
      derived_1(int ii, int jj, int i_bas, int j_bas);
      ~derived_1() { cout << " destruktor derived_1\n"; }
      int get_id() { return id; }
      void show() {
            cout << "derived_1 : id = " << id << " jd = " << jd << endl;
            base::show();
      }
};

//implementation of class derived_1

derived_1::derived_1(int ii, int jj, int i_bas, int j_bas) : base(i_bas, j_bas)
```

```
/*=============================================================================
Konstruktor klasy derived_1
=============================================================================*/
{
	id = ii;
	jd = jj;
	cout << "konstruktor derived_1\n";
}


//declaration of derived_2
class derived_2 : public derived_1
{
	int id;
public:
	int jd;
	derived_2(int ii, int jj, int ii_d1, int jj_d1, int i_bas, int j_bas);
	~derived_2() { cout << " destruktor derived_2\n"; }
	int get_id() { return id; }
	void show() {
		cout << "derived_2 : id = " << id << " jd = " << jd << endl;
		derived_1::show();
	}
};

//implementation of class derived_2

derived_2::derived_2(int ii, int jj, int ii_d1, int jj_d1, int i_bas, int j_bas)
                                                        : derived_1(ii_d1, jj_d1, i_bas, j_bas)
/*=============================================================================
Konstruktor klasy derived_1
=============================================================================*/
{
	id = ii;
	jd = jj;
	cout << "konstruktor derived_2\n";
}

int _tmain(int argc, _TCHAR* argv[])
{
	derived_2 d2(500, 600, 50, 60, 5, 6);
	d2.show();
```

```
	system("pause");

	return 0;
}
```