

Szoftver laboratórium II. 3. ellenőrző dolgozat. 2013.04.09. Kurz/Terem: L4/	15 perc
Név:	Neptun:
	Összpont:

1. feladat

6 pont

A **pontozott** helyek felhasználásával **egészítse ki** az alábbi programot úgy, hogy a főprogram kommentezett sorai az **előre megadott** kimenetet (1 2,3 2) adják! Ügyeljen a tagváltozók elérhetőségére! (3p)

Ezt követően a főprogram minden sorához is adja meg, hogy az mit ír ki a szabványos kimenetre! (3p) Jelölje a szóköz karaktereket is! (Nem biztos, hogy minden megadott helye írnia kell!)

```
#include <iostream>
using namespace std;
class Pont1 {
    int x;          // x koordináta
public:
    Pont1(int x = 0)
        : x(x)
        { cout << "P1ctor "; }
    Pont1(const Pont1& p)
        : x(p.x)
        { cout << "P1copy "; }
    /*.....*/ void print(ostream& os) const { os << x; }
    virtual ~Pont1()
        { cout << "P1dctor "; }
};
class Pont2 :public Pont1 {
    int y;          // y koordináta
public:
    Pont2(int x = 0, int y = 0)
        :Pont1(x), y(y)
        { cout << "P2ctor "; }
    Pont2& operator=(const Pont2& p2)
        { Pont1::operator=(p2); y = p2.y; cout << "P2= "; return *this;}
    ~Pont2()
        { cout << "P2dctor "; }
    /* ..... */
    void print(ostream& os) const {
        Pont1::print(os);
        os << ", " << y;
    }
    /* ..... */
};
/* ..... */
ostream& operator<<(ostream& os, const Pont1& p) {
    p.print(os);
    return os;
}
/* ..... */

int main() {
    Pont1 p1(1);      cout << endl;    // P1ctor _____
    cout << p1        << endl;    // 1
    Pont2 p23(2,3);  cout << endl;    // P1ctor P2ctor _____
    Pont2 p20;       cout << endl;    // p1ctor P2ctor _____
    p20 = p23;       cout << endl;    // P2= _____
    p20.print(cout); cout << endl;    // 2,3
    Pont1 *pp = &p20; cout << endl;    // _____
    pp->print(cout); cout << endl;    // 2
    return 0;        // P2dctor P1dctor P2dctor P1dctor P1dctor _____
}

```