

Szoftver laboratórium 2. 2. ellenőrző dolgozat. 2011.03.12. Kurz/Terem: L1/	15 perc
Név: _____ Neptun: _____	Összpont: _____

1.feladat

6 pont

Valósítson meg C++ nyelven egy olyan osztályt (**Vektor**), ami 2D vektor tárolására alkalmas! Az osztály tárolja a vektor valós koordinátáit (x,y) és valósítsa meg a következő műveleteket!

- összeadás (+)
- kiírás egy ostream típusú objektumra a szokásos << operátorral. A kiírás formátumát Ön határozza meg, és ezt adja meg az alábbi programrészletben. (A vonalakra írja rá, hogy mit ír ki! Jelölje a szöközőket is!)

A vektor koordinátái kívülről közvetlenül ne legyenek elérhetőek! Működjön helyesen az alábbi kódrészlet, de **ne írjon felesleges** tagfüggvényt (olyat, ami automatikusan keletkezik és jó a feladathoz)!

```

...
const Vektor v00; // Ez egy nullvektor
const Vektor v10(1.0); // Ennek az x koordinátája 1, az y nulla
const Vektor v11(1,1); // Ennek minden koordinátája 1
Vektor v0 = v00;
Vektor *vp = new Vektor[10];
std::cout << vp[1] << std::endl; // (0,0) _____
for (int i = 0; i < 10; i++)
    vp[i] = vp[i] + v10;
vp[0] = v00 + v10;
std::cout << vp[0] << std::endl; // (1,0) _____

```

Hányszor hívódott meg a fenti kódrészletben a **paraméter nélküli**<sup>1</sup> konstruktor? **11**, (az operator+ is hasznáta volna, akkor 12)

```

class Vektor {
    double x;
    double y;
public:
    Vektor(double x = 0, double y = 0) :x(x), y(y) {}
    Vektor operator+(const Vektor&) const;
    friend ostream& operator<<(ostream&, const Vektor&);
};

```

```

Vektor Vektor::operator+(const Vektor& rv) const {
    return Vektor(x + rv.x, y + rv.y);
}

```

```

ostream& operator<<(ostream& os, const Vektor& rv) {
    os << "(" << rv.x << "," << rv.y << ")";
    return os;
}

```

<sup>1</sup> A kérdés a paraméter nélkül **hívható** (default) konstruktorra vonatkozott. Paraméter nélküli konstruktora az osztálynak valójában nincs, mivel ott vannak a default argumentumok. Aki az utóbbi formában értelmezte a kérdést annak a 0-át is elfogadtuk válaszként.