

| | |
|---|-----------|
| Programozás alapjai 2. 1. ellenőrző dolgozat. 2014.03.13. Kurz/Terem: G3/ | |
| Név: | Összpont: |

1. feladat

3.5 pont

Valósítson meg C++ nyelven egy egyszerű osztályt (*UpTime*), ami egy rendszerben a bekapcsolás óta eltelt másodperceket (**long int**) tárolja! Legyen az osztálynak:

- olyan paraméter nélküli konstruktora, amivel a bekapcsoláskor létrehozható az objektum;
- olyan destruktora, ami szabványos kimenetre kiírja a tárolt időt másodpercekben;
- olyan tagfüggvénye (*tick*), amit a rendszer minden másodpercben meghívhat és növeli a tárolt időt;
- olyan ++ operátora, ami *tick* tagfüggvényéhez hasonlóan növeli az időt (Ön dönti el, hogy melyik ++ operátort használja és azt is, hogy az ad-e vissza értéket);
- olyan tagfüggvénye (*getTime*) amivel le lehet kérdezni a tárolt időt!

Az osztálynak legyen legalább egy nem inline tagfüggvénye! Az osztály adatait közvetlenül ne lehessen módosítani!

```
class Uptime {
    long time;
public:
    Uptime() :time(0) {}
    ~Uptime() { std::cout << time; }
    void tick() { ++time; }
    Uptime operator++();
    long getTime() const { return time; }
};
```

```
Uptime Uptime::operator++() {
    tick();
    return *this;
}
```

2. feladat

2.5 pont

Írja a vonalakra, hogy mit ír ki az alábbi C++ program a standard outputra! Jelölje a szóközöket is (nem biztos, hogy minden vonalra kell írni)!

```
#include <iostream>
using namespace std;
class Zh1 {
    char a;
public:
    Zh1() :a('A')           { cout << 'k'; }
    Zh1(const Zh1& zh) :a(zh.a) { cout << 'c'; }
    ~Zh1()                 { cout << 'd'; }
};
ostream& operator<<(ostream& os, const Zh1&) { os << 'o'; return os;}
int main() {
    Zh1 v1;                cout << endl; // k _____
    Zh1 v2 = v1;           cout << endl; // c _____
    Zh1 v3[2];             cout << endl; // kk _____
    cout << v1 << endl;     // o _____
    return 0;              // dddd _____
}
```