

Szoftver laboratórium II. 4. ellenőrző dolgozat. 2013.04.23. Kurz/Terem: L2/	15 perc
Név: _____ Neptun: _____	Összpont: _____

Egy **biztonsági rendszerben** különböző eseményeket (betörés, áramszünet, tűz, stb.) heterogén gyűjteményként szeretnénk tárolni. Ehhez elkészítettük az *Obj* alapsztályt, ami eltárolja az aktuális időpontot (az objektum létrejöttének időpontját az *Ido* osztály konstruktora állítja elő), a *print()* tagfüggvénye pedig képes kiírni azt a standard kimenetre.

```
class Obj {
    Ido idopont; // esemény keletkezésének (észlelésének) időpontja
public:
    virtual void print() const {
        cout << idopont;      } // kiírja az észlelés időpontját
    virtual ~Obj() {}
};
```

Az *Obj* osztály felhasználásával megterveztünk néhány konkrét eseményosztályt:

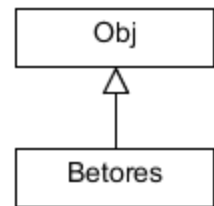
Esemény neve	Konstruktork	Jellemző adat
betörés	Betores(string hely)	hely: string
áramszünet	SzUNET(int hossz)	áramszünet hossza: int
tűz	Tuz(double terület)	terület nagysága: double

1. feladat

3 pont

Az Ön feladata *Betores* osztály megvalósítása, amely a betörés helyét tárolja. **Deklarálja és implementálja** (valósítsa meg) az osztályt úgy, hogy annak *print()* tagfüggvénye írja ki a standard kimenetre az esemény nevét, az észlelés idejét, és az eseményre jellemző adatot (helyet). UML ábrával mutassa be a *Betores* és az *Obj* osztály kapcsolatát! Elegendő az osztályok neveit feltüntetni, az adattagokat és függvényeket nem kell.

```
class Betores : public Obj {
    string hely;
public:
    Betores(string hely) : hely(hely) {}
    void print() const {
        Obj::print();
        cout << " Betores helye: " << hely;
    }
};
```



2. feladat

3 pont

Tételezze fel, hogy az előző feladat eseményosztályai elkészültek! Az előző laborgyakorlaton elkészített generikus tároló (*Array*: indexelhető, van iterátora, és van *size()* tagfüggvénye is) **felhasználásával** írjon **kódrészletet**, amelyben létrehoz egy eseménytárat, maximum 600 különböző esemény tárolására, és tároljon el benne 80 betörés (*Betores*) eseményt, valamint 50 tűz (*Tuz*) eseményt! (Az események jellemző adatának bármit megadhat!)
Készítsen függvénysablont (*listaz*), amely meghívja egy heterogén kollekció két iterátora (*first*, *last*) közötti összes esemény *print()* tagfüggvényét! Mutassa be a függvénysablon használatát az imént létrehozott tárolón!

```
template <class Iter>
void listaz(Iter first, Iter last){
    while (first != last) {
        (*first++)->print();
        cout << endl;
    }
}

Array<Obj*, 600> t;
for (size_t i = 0; i < 80; i++) t.at(i) = new Betores("Ott");
for (size_t i = 80; i < 130; i++) t.at(i) = new Tuz(1.43);
listaz(t.begin(), t.end());
```