

Szoftver laboratórium 2.	3. Ellenőrző dolgozat.	2014.03.31.	Kurz/Terem: L1/
Név:	Neptun:	Összpont:	

1.feladat

2 pont

Tervezzon egy olyan osztályt (*Mosopor*), ami képes egy mosópor nevét tárolni! A konstruktorban legyen megadható a mosópor neve, a paraméter nélkül hívott konstruktor pedig üres nevet állítson be! A név legyen lekérdezhető! A tetszőleges hosszúságú név tárolásához használja fel a korábbi laborgyakorlatokon elkészült *String* osztályt! Osztálydiagram segítségével ábrázolja a két osztály kapcsolatát! A *String* osztály tagfüggvényeit és adatait nem kell részleteznie a rajzon.

Valósítsa meg C++ nyelven a *Mosopor* osztályt!

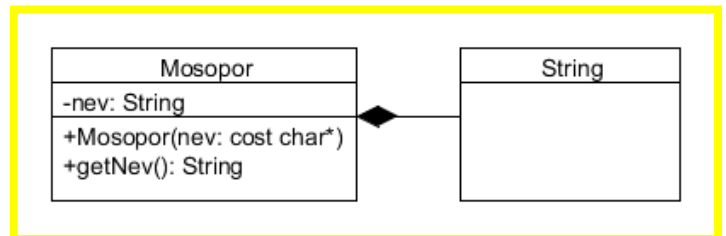
2.feladat

4 pont

Tervezzon egy olyan tárolót (*Raktar*), ami tetszőleges számú *Mosopor* típusú objektumot képes tárolni! Az osztálynak legyen olyan tagfüggvénye, amivel új mosóport tehetünk a raktárba (*berak*), és olyan is, amivel lekérdezhethetjük, hogy hány mosóport van a raktárban (*keszlet*)! Elvárás az osztállyal szemben, hogy másolható legyen, értékadás bal és jobb oldalán is szerepelhessen.

Deklarálja C++ nyelven a *Raktar* osztályt! Valósítsa meg (definiálja) az osztály paraméter nélkül hívható konstruktorát, destruktort, valamint az értékadó operátort! Az alábbi kódrészlet az osztályok használatára mutat példát:

```
....
Mosopor egyik("Jo por"), masik("meg jobb por");
Raktar budaorsi, csepeli;
budaorsi.berak(egyik);
budaorsi.berak(masik);
csepeli = budaorsi;
std::cout << csepeli.keszlet();
```



```
class Mosopor {
    String nev;
public:
    Mosopor(const char* n = "") :nev(n) {}
    String getNev() const { return nev; }
};
```

```
class Raktar {
    Mosopor *mp;
    int db;
public:
    Raktar() :db(0), mp(0) {}
    Raktar(const Raktar&);
    int keszlet() const;
    void berak(const Mosopor&);
    Raktar& operator=(const Raktar&);
    ~Raktar() { delete[] mp; }
};
```

```
Raktar& Raktar::operator=(const Raktar& rhs) {
    if (this != &rhs) {
        delete[] mp;
        db = rhs.db;
        mp = new Mosopor[db];
        for (int i = 0; i < db; i++)
            mp[i] = rhs.mp[i];
    }
    return *this;
}
```