

## Programozás II. gyakorlati vizsga példa

Készítsünk egy egyetemet! Legyen egy **Hallgató** osztály, amely a hallgató nevét (*szöveg*), EHA kódját (*szöveg*), h-s azonosító (*egész szám*), valamint a képességét (*előjeltelen egész*) tárolja.

- Legyen paraméteres konstruktora, ami az adattagokat várja paraméterül, és beállítja azokat.
- Legyen felüldefiniálva a `--` operátor, amely egy számot vár, és a hallgató képességét csökkenti a paraméterben megadott számmal.
- Lehessen kiírni ostream osztály példányokra az `operator <<` segítségével

Készíts egy **Dolgozat** osztályt is, amely a dolgozaton elérhető maximális pontszámot, és a dolgozat nehézségét tárolja. Legyen neki paraméteres konstruktora, illetve legyen felüldefiniálva a `<<` operátor.

Készíts egy **Egyetem** osztályt, amely Dolgozatokat és Hallgatókat tárolja dinamikus tömbben.

- Legyen paraméteres konstruktora, amely a dolgozatok és a hallgatók maximális számát várja.
- Legyen két `operator+=` függvénye. Egyik Dolgozat, másik Hallgató típusú paramétert vár. Amennyiben lehetséges, elhelyezi a kapott paraméter másolatát (copy konstruktor segítségével) a megfelelő tömbben. Ha nincs már több hely, dobjon kivételt.
- Legyen egy `operator<<` függvénye, mely kiírja a hallgatókat, a dolgozatokat, valamint hogy hány üres hely van az egyes tömbökben.
- Legyen egy `dolgozatMegir(int)` metódus, mely a dolgozatok közül a paraméterként kapott sorszámú dolgozatot „megíratja” a hallgatókkal. Írassuk ki az alapértelmezett kimenetre minden hallgató eredményét, az alábbi képlet segítségével:  $pontszám = (képesség / nehézség) * pontszám$ . Ha a képlet magasabb eredményt ad, mint a dolgozat maximális pontszáma, a hallgató a maximális pontszámot kapja.

Legyen az Egyetem osztálynak felüldefiniálva a `--` operátora, amely a hallgató EHA kódját várja paraméterül, és kitörli az adott EHA kódú felhasználót.

Legyen az Egyetem osztálynak egy `operator[]` függvénye, mely a paraméterül kapott h-s azonosítójú hallgatóval tér vissza.

**A main függvényben hozz létre Egyetemet, és próbáld ki a megírt függvényeket!**

### ***Figyelj oda a***

- memória felszabadításra
- másoló konstruktorra, destruktorra és értékadás operátor felüldefiniálásra (csak ahol szükséges, tehát olyan osztályoknál, melyeknek van dinamikus adattagja!)
- láthatóságok használatára

Jó munkát!