

Írj osztályt, ami egy *Harcos* objektumot valósít meg.

- A *Harcos* adattagjai a következők legyenek: név (String), életero (int), harci ero (int). Az adattagok csak ebből az osztályból legyen elérhetőek.
- Készíts az osztályhoz konstruktor, ami paraméterek alapján állítja be az adattagokat.
- Az osztálynak legyen egy boolean *harcol* metódusa, ami egy másik harcost kap paraméternek. A metódus mindkét *harcos* életerejét csökkentse a másik *harcos* harci erejével.
- Ha valamelyik *harcos* elveszti a harcot (életerejé 0 alá csökkenne), a metódus térjen vissza igazgal, egyébként hamissal.
- Készítsd el a megfelelő metódusokat az adattagok lekérdezéséhez és módosításához.
- Készíts *toString* metódust az osztályhoz.
- A *main* metódusban hozz létre két *harcos* objektumot. *Harcoljanak*, amíg valamelyikük el nem veszti a harcot.

Írj osztályt, ami egy *Macska* objektumot valósít meg.

- A *macska* adattagjai a következők legyenek: név (string), súly (double), éhes -e (boolean).
- Két konstruktort is készíts az osztályhoz. Az egyik általános legyen, ami minden adattagot a konstruktor paraméterlistájából állít be, illetve egy másik, ami az első két adattagot a konstruktor paraméterlistájából kapja, és alapértelmezetten éhes a macska legyen.
- Az osztálynak legyen egy *eszik* metódusa, ami egy double értéket vár (étel mennyisége), és egy boolean-el tér vissza (sikeres volt -e az etetés). Ha a macska éhes, az etetés sikeres, és a súlya nőjön az étel mennyiségével. A macska ezután ne legyen éhes. Ha a macska nem éhes, az etetés nem sikeres.
- Az osztálynak legyen egy void *futkos* metódusa, ami nem vár paramétert. A macska súlya csökkenjen 0.1-el, és ha nem volt éhes, akkor éhezzen meg.
- Készíts *toString* metódust az osztályhoz.
- A *main* metódusban hozz létre két macskát a két különböző konstruktorral, és próbáld meg megetetni őket. Az etetés sikerességéről írd ki információt konzolra.
- Mindkét macska futkosson, és utána írd ki szövegesen az objektumokat.

Írj osztályt, ami egy *Számítógép* objektumot valósít meg.

- A számítógép adattagjai a következők legyenek: szabad memória MB-ban (double), be van -e kapcsolva (boolean).
- Készíts két konstruktort is az osztályhoz. Az egyik általános legyen, ami minden adattagot a paraméterlistából állít be, a másik egy alapértelmezett konstruktor legyen, ami 1024 MB memóriával, kikapcsolva hozza létre a gépet.
- Az osztálynak legyen egy void *kapcsol* metódusa, ami nem vár paramétert. Ha a gép ki van kapcsolva, akkor kapcsolja be, egyébként kapcsolja ki.
- Az osztálynak legyen egy boolean *programMasol* metódusa, ami egy program méretét várja paraméternek MB-ban (double). Ha a program ráfér még a gépre, és a gép be van

kapcsolva, úgy csökkenjen a szabad memória a program méretével. A metódus térjen vissza boolean változóval, hogy sikeres volt -e a másolás.

- Készíts toString metódust az osztályhoz.
- A *main* metódusban hozz létre két számítógépet a fenti konstruktorokkal. Mindkét gép kikapcsolt állapotban kezdjen. Az alapértelmezett gépet kapcsold be, és másold rá eloször 800 MB, aztán 400 MB programot. A másik gépre másolj 1 MB programot. A másolások eredményeit írd ki.
- Mindkét objektumot írd ki szövegesen.

Írj osztályt, ami egy *Hallgato* objektumot valósít meg.

- A hallgató adattagjai a következők legyenek: azonosító (string), évfolyam (int), kreditszám (int).
- Két konstruktort is készíts az osztályhoz. Az egyik általános legyen, ami minden adattagot a konstruktor paraméterlistájából állít be, illetve egy másik, ami az első adattagot a konstruktor paraméterlistájából kapja, évfolyama 1 és kreditszáma 0 legyen.
- Az osztálynak legyen egy void tárgyFelvesz metódusa, amivel egy int paramétert (tárgy kreditértéke) kér. A hallgató kreditszáma nőjön a kapott értékkel.
- Az osztálynak legyen egy boolean vizsgazik metódusa, ami nem vár paramétert. Ha a hallgatónak 0-nál több kreditje van, akkor a sikeres a vizsga: a következő évfolyamba lép, és nullázódik a kreditszáma. Egyébként a vizsga sikertelen.
- Készíts toString metódust az osztályhoz.
- A main metódusban hozz létre két hallgatót a két különböző konstruktorral. Az egyikük vegyen fel tárgyat, majd vizsgáztasd őket. A vizsga sikerességéről írd ki információt konzolra.
- Ezután mindkét hallgatót írd ki szövegesen.