

Gyakorló feladatok 2024. Április 3.

Összetett programozási tételek

1. Ismerd meg a másolás programozási tételt és a tétel pszeudokódját.
2. Készítsünk egy **ProgTetel** nevű osztályt. Ennek az osztálynak legyen egy publikus és statikus (azaz példányosítás nélkül meghívható) **int[] TombDuplazo(int[] x)** metódusa, ami a paraméterül kapott tömb értékeit átmásolja a kimeneti tömbbe, úgy, hogy az értékeket megkétszerezi. Ellenőrizd a kódod működését!
3. Ismerd meg a kiválogatás programozási tételt és a pszeudokódját.
4. Készíts a **ProgTetel** osztályba egy publikus **int[] ParostKivalogat(int[] x)** statikus metódust, amely létrehoz egy eredmény tömböt, amit feltölt csupa -1 -el. Majd a paraméterül kapott tömbből kiválogatja a páros számokat és azt a tömb elejére helyezi (azt adja eredményül). Ellenőrizd a kódod működését!
5. Ismerd meg a kiválogatás helyben programozási tételt és a pszeudokódját!
6. Készítsd el a **ProgTetel** osztályba a 4. feladat megoldását kiválogatás helyben programozási tétel segítségével. A metódus neve **ParostHelybenKivalogat** legyen! Ellenőrizd a kódod működését!
7. Ismerd meg a szétválogatás (két tömbbe) programozási tételt és a pszeudokódját.
8. Készítsük a **ProgTetel** osztályba egy publikus **void SzamokatSzetValogat(int[] tomb, ref int[] paros, ref int[] paratlan, ref int db1, ref int db2)** statikus metódust. A paraméterül kapott tomb elemeit válogassa szét páros és páros számokat tartalmazó tömbbe. A referencia szerint átadott tömbökbe történjen a szétválogatás!
A db1 tartalmazza a páros számok számát, míg a db2 a páratlan számok számát. Ellenőrizd a kódod működését!
9. Ismerjük meg a szétválogatás helyben programozási tételt és a pszeudokódját.
10. Készítsük el a 8. feladat metódusát szétválogatás helyben programozási tétel segítségével is! A metódus neve: **SzetvalogatHelyben** legyen! Ellenőrizd a kódunk működését!

További feladatok (lehetőleg önállóan):

Készítsünk egy **Személy** nevű osztályt, ami egy csoporttársról tárolja az alábbi adatokat:

- *név* (string),
- *nem* (bool) false értéknél leány, true értéknél fiú,
- *progjegy* (int) - előző félévi prog jegye.

A mezők értékét konstruktorban lehessen beállítani, valamint egy-egy tulajdonsággal lehessen lekérdezni.

Készítsünk egy **Kurzus** osztályt, ami majd Személy-eket fog tárolni. Legyen egy belső tömbje, ami a Személyeket tárolja, ennek méretét a konstruktorban lehessen megadni. A kurzusra maximum 24 fő jelentkezhet, ennél nagyobb szám esetén dobjon egy **MaxLetszamException()** kivételt.

Legyen két metódusa:

- **bool Felvétel(Személy sz)**, ami a paraméterül kapott személyt elhelyezi a tömbben (ha lehetséges).
- **Bool Töröl(Személy sz)**, ami a paraméterül kapott személyt törli a tömbből.

Készítsünk egy `Kurzus` és pár `Személy` objektumot és vegyük fel a `Kurzus`-ra a `Személy`eket.

Programozási tételek segítségével oldjuk meg az alábbi feladatokat (a feladatok leírása alatt a megvalósítandó metódus nevét látod). Minden metódust a `Kurzus` osztályban valósítsunk meg!

A kurzuson hány leány van?

```
int GirlsNumber();
```

Van-e a kurzuson olyan fiú-lány páros, akiknél mindkettőjük prog jegye 5 volt?

```
bool GeekCouple();
```

Válogassuk ki az ötös prog jeggyel rendelkező `Személy`eket!

```
Személy[] ProProgrammers();
```

Legyen egy szétválogató metódusunk ami a belső tömb elejére teszi a lányokat, majd utána a fiúkat.

```
void OrderbyGender();
```

Készítsünk tesztekkel amikkel ellenőrizni tudjuk a metódusaink működését.