

LUFTHANSA

A Poggyász-Tetris



Helyszín: Budapest, Lufthansa Systems – Innovációs Központ, VIP Prezentációs Terem.

Időpont: 13:30

A hatalmas üveglablakokon át látni a magasban szálló repülőgépek, de a teremben most más vonja magára a figyelmet. A 11.c osztály céglátogatáson vesz részt a Lufthansa Systems-nél, és a bemutatót tartó vezető fejlesztő enyhén izzadva lazítja meg a nyakkendőjét. Az osztály ugyanis letarolta a termet: záporoznak a kérdések a szerverparkok optimalizálásáról, az adatszerkezetekről és a hálózati terheléelosztásról. Olyan zseniális meglátásaik vannak, hogy a terem végében csendben figyelő Igazgató is közelebb lép.

Az Igazgató megköszörüli a torkát, és elmosolyodik:

– „Látom, az elmélet nagyon megy. Mit szóltok egy éles teszthez? Holnap reggel indul egy különgépünk Costa Ricára. A földi kiszolgáló rendszerünk poggyászelosztó modulja viszont reggel óta hibát jelez, és 300 csomag vár a szalagon. Ha a 11.c képes egy működő C# algoritmussal elosztani a csomagokat úgy, hogy a gép súlypontja tökéletes maradjon, a cég állja az egész osztály repülőjegyét arra a Costa Rica-i járatra. Áll az alku?”

A küldetésed

A tét hatalmas: a tengerpart és a dzsungel vár, de csak akkor, ha a gép biztonságosan fel tud szállni. A fegyvered a C# és a programozói logikád. A feladatod, hogy a szalagon érkező csomagokat úgy oszd el a repülőgép 5 darab ULD (Unit Load Device) konténerében, hogy a fizika és a logisztika szabályai is teljesüljenek.

Dömping

A Család-szabály: A gyors poggyászkidás érdekében egy család (azonos FamilyID) összes csomagjának szigorúan **ugyanabba a konténerbe** kell kerülnie! Nem szakíthatjátok szét egy család csomagjait. Ezért az adatokat a fájlból egy Dictionary (Szótár) adatszerkezetbe kell beolvasnod és csoportosítanod a Család azonosítója alapján. *Vigyázz! A struct érték típusú (value type). Ha egy szótárból lekérsz egy struktúrát, megváltoztatod a súlyát (mert hozzáadsz egy újabb csomagot), majd elfelejtetd visszatenni a szótárba, az adatok elvesznek! Figyelj az adatok felülírására!*

1. Feladat: Olvassa a be **csomagok.csv** fájl tartalmát és tárolja le a meghatározott adatszerkezetben.

- a. Hozzon létre egy struktúrát Csomag megnevezéssel, ami 3 információt hordoz: azonosító (szöveg), súly (valós szám), térfogat (valós szám)
- b. Hozzon létre egy struktúrát csaladiCsomagok névvel, ami a tartalmazza a család azonosítót (string), összsúly (valós szám), össztérfogat (valós szám) és egy Csomag típusú csomagok (lista).
- c. A csomagok.csv fájl sorai a következőképpen épülnek fel: csomag azonosító;család azonosító;súly (kg);térfogat (m³)
- d. Az adatokat `Dictionary<string, CsaladiCsomagok>` adatszerkezetbe tárold le!

[2]

Egyensúly (Balance): A repülőgép 5 konténerének pozíciója a gép súlypontjához képest (ezt hívjuk Erőkarnak) a következő:

- Konténer 1 (Orr): -2
- Konténer 2: -1
- Konténer 3 (Közép): 0
- Konténer 4: +1
- Konténer 5 (Farok): +2

Kapacitás limit: Egy konténer maximum **1500 kg** súlyt és maximum **6.0 m³** térfogatot bír el. Ezt túllépni szigorúan tilos!

2. Feladat: írd egy függvényt, amely inicializálja (beállítja a kezdőértékeket) a konténereket:

- a. Hozzon létre egy konténer struktúrát, ami a következő adatokat tárolja le: azonosító (egész), erőkar (egész), jelenlegi súly (valós szám), jelenlegi térfogat (valós szám), betöltött családi csomagok (Lista<CsaladiCsomagok>)

- b. Hozzon létre egy függvényt, ami inicializálja és visszatér az 5 hosszúságú konténer tömbbel a leírás alapján. Kezdetben mind az 5 konténer üres
- c. Hozzon létre egy logikai függvényt, ami eldönti, hogy az adott konténerbe az adott csomag még befér-e. [3]

A cél, hogy a repülőgép teljes Súlypontja (CG - Center of Gravity) a lehető legközelebb legyen a 0-hoz. A fizikai képlet, amit a kódba is be kell építened, a következő:

$$CG = \frac{\sum (Konténer\ Súly \times Erőkar)}{\sum Konténer\ Súly}$$

3. **Feladat:** Írj egy függvényt, amely meghatározza a K-adik legnehezebb csomagot.
4. **Feladat:** Írj egy függvényt, ami meghatározza a CG-t, ami a konténer súlyát az erőkar figyelembevételével súlyozva átlagolja! [4]

A Mohó algoritmus

1. **Mérlegelés:** Menj végig a családokon. Minden családnál nézd meg mind az 5 konténert:
 - Befér-e a súly és a térfogat?
 - Ha igen, számold ki, hogyan változna a gép súlypontja (CG), ha betennéd.
 - Pakold be a csomagokat abba a konténerbe, amelyik a súlypontot a legközelebb húzza a tökéletes 0-hoz!
2. Írd ki a minta szerint a feltöltés eredményét, ha a súlypont kevesebb, mint 0.5-tel tér el, akkor felszállhat a gép! [5]

```
--- LUFTHANSA JÁRAT RAKODÁSI TERV ---
Konténer 1 (Erőkar: -2): 1043,70 kg / 5,77 m^3 - 29 család
Konténer 2 (Erőkar: -1): 982,40 kg / 5,98 m^3 - 17 család
Konténer 3 (Erőkar: 0): 1004,70 kg / 5,94 m^3 - 11 család
Konténer 4 (Erőkar: 1): 1001,10 kg / 5,96 m^3 - 17 család
Konténer 5 (Erőkar: 2): 1021,10 kg / 5,71 m^3 - 29 család

A repülőgép VÉGSŐ súlypontja (CG): -0,0052
A gép tökéletes egyensúlyban van. Felszállás engedélyezve!
```