

**1. feladat:**  
**Programozás: Derékszögű háromszögek**

**Összesen: 60 pont**

**1.1. A következő feladat megoldásához hozzon létre grafikus alkalmazást (projektet) *Haromszögek* azonosítóval!**

**1.2. Hozzon létre saját osztályt *DHaromszog* azonosítóval a következő osztálydiagram szerint!**

A „lakattal” jelölt mezők és jellemzők privátak, a többi tag publikus legyen!

**1.3. Készítse el az osztály konstruktorát (*DHaromszog (...)*) is!**

A konstruktor *sor* nevű paramétere string típusú, sorrendben egy derékszögű háromszög „a”, „b” befogóit és „c” átfogóját kapja értékül, például:

„1,11 1,48 1,85”

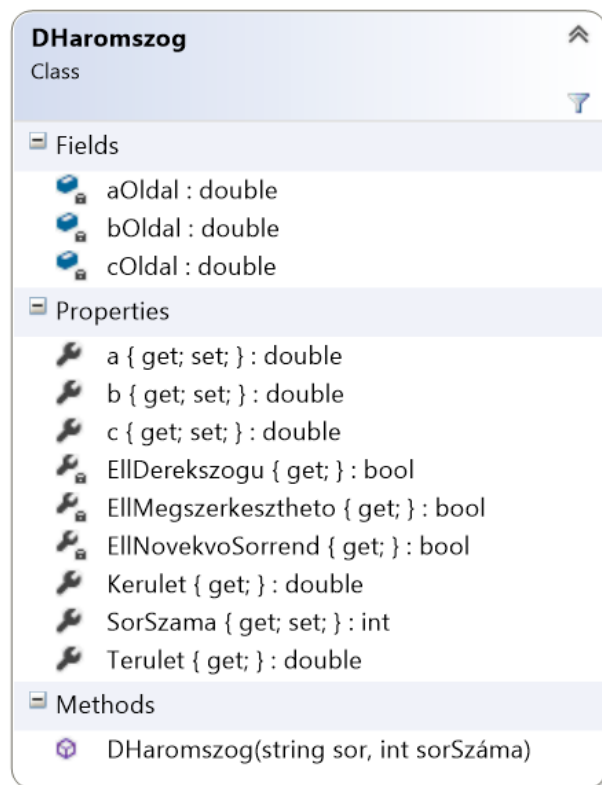
A valós értékeket szóköz karakterek választják el egymástól. A háromszög oldalai a 4. feladatban leírtak szerint kapjanak értéket a publikus jellemzőkön keresztül!

A *sorSzama* paraméter és a kapcsolódó jellemző az oldalak sorszámát rögzítik az input (*haromszögek.txt*) fájlban. A fájl sorait 1-től kezdődően számozzuk.

A konstruktorban a következő ellenőrzéseket kell csak elvégeznie:

- Az inputadatokra teljesülni kell az  $a \leq b \leq c$  feltételnek.
- A háromszögnek megszerkeszthetőnek kell lennie ( $a + b > c$ ).
- A háromszögnek derékszögűnek kell lennie ( $c^2 = a^2 + b^2$ ).

A fenti feltételek ellenőrzésére készítsen csak olvasható privát jellemzőket! Ha a feltételek nem teljesülnek, akkor a konstruktor dobjon saját kivételt a következő oldalon található minta szerinti üzenettel!



```

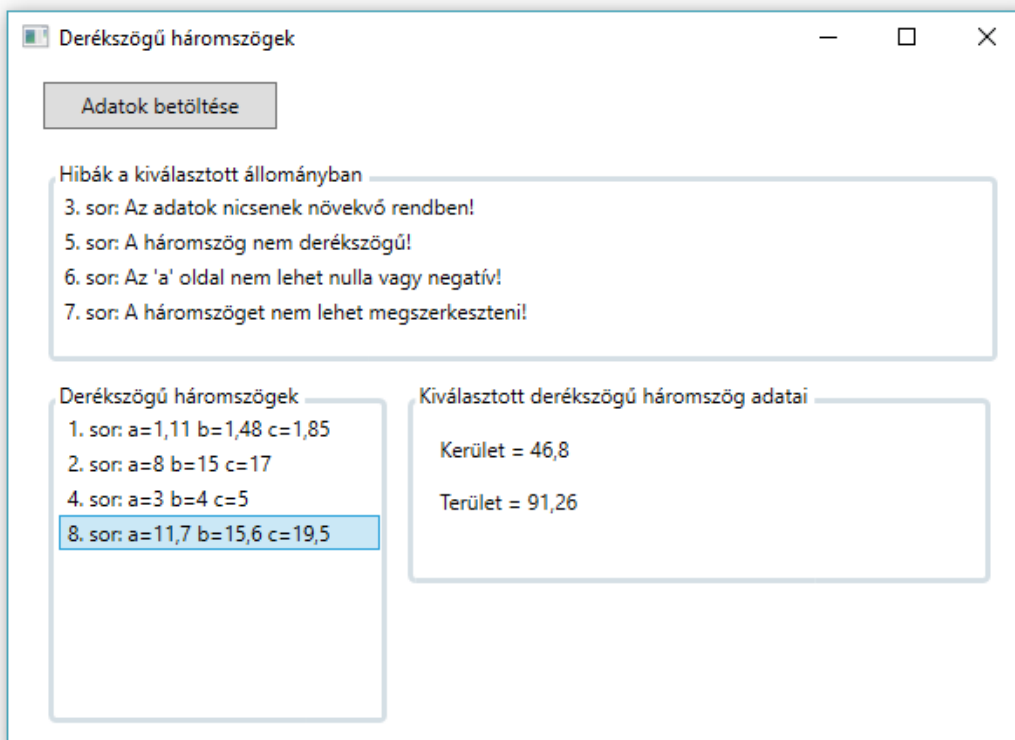
DHaromszog
Class
  Fields
    aOldal : double
    bOldal : double
    cOldal : double
  Properties
    a { get; set; } : double
    b { get; set; } : double
    c { get; set; } : double
    EllDerekszogu { get; } : bool
    EllMegerszerkesztheto { get; } : bool
    EllNovekvoSorrend { get; } : bool
    Kerulet { get; } : double
    SorSzama { get; set; } : int
    Terulet { get; } : double
  Methods
    DHaromszog(string sor, int sorSzama)
  
```

- 1.4. A publikus „a”, „b”, „c” jellemzők (Properties) csak akkor módosítsák a privát mezők (Fields) „aOldal”, „bOldal”, „cOldal” értékét, ha a tárolandó érték nagyobb mint nulla!**

Ha a feltétel nem teljesül, akkor a jellemző dobjon saját kivételt az értékadás során a „A(z) '...' oldal nem lehet nulla vagy negatív!” üzenettel!

- 1.5. Készítsen publikus, csak olvasható jellemzőket a terület ( $a * b / 2$ ) és a kerület ( $a + b + c$ ) meghatározásához!**

- 1.6. Készítsen grafikus felhasználói felületet a háromszögek kezeléséhez a következő minta alapján!**



- 1.7. Az „Adatok betöltése” parancsgombra kattintva legyen kiválasztható a beépített fájl megnyitása párbeszédablakkal egy szöveges állomány, melyben a háromszögek oldalainak hosszúságát rögzítettük! A szöveges fájlban található adatsorok felépítése a 3. feladatban leírtak szerint történjen! Programja teszteléséhez használja a *haromszokek.txt* forrásállományt! Ha sikeres volt egy állomány kiválasztása, akkor törölje a szükséges változók tartalmát (listákat, vektorokat stb.), hogy az előzőleg kiválasztott állomány adatai ne jelenjenek meg a grafikus felületen!**

- 1.8. Tárolja el egy *DHaromszog* típusú összetett adatszerkezetben (vektorban, listában stb.) a kiválasztott szöveges állományban található háromszögek adatait (*DHaromszog* osztálpéldányokat)! Csak a derékszögű háromszögek adatait tárolja el! A beolvasás során keletkezett hibákat try-catch szerkezettel kezelje, és a hibákhoz tartozó üzenetek jelenjenek meg a mintán látható listában!**

**1.9. A sikeresen tárolt derékszögű háromszögek adatai jelenjenek meg a minta szerinti listában a bal oldalon!**

**1.10. A „Derékszögű háromszögek” listában kiválasztott háromszög kerülete és területe jelenjen meg a minta szerint a jobb oldalon!**