

--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

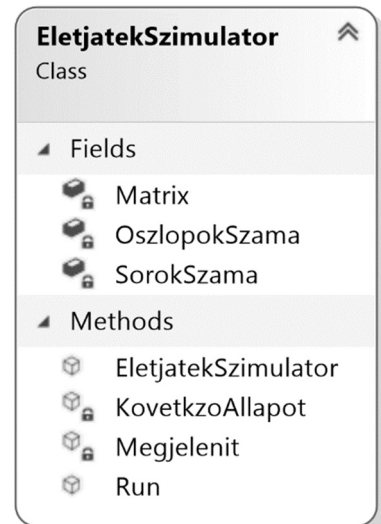
2. feladat

40 pont

Életjáték szimulátor¹

Az életjátékot John Horton Conway, a Cambridge-i Egyetem matematikusa találta ki. Játékként való megnevezése megtévesztő lehet, mivel „nullszemélyes” játék. A „játékos” szerepe mindössze annyi, hogy megad egy kezdőalakzatot, és azután csak figyeli az eredményt. Matematikai szempontból az ún. **sejtautomaták** közé tartozik. A játék egyes lépéseinek eredményét számítógép számítja ki. Ebben a feladatban egy életjáték szimulátort kell készítenie.

- Készítsen **konzolos alkalmazást** a következő feladatok megoldására, melynek projektjét Eletjatek néven mentse el!
- Készítsen saját osztályt EletjatekSzimulátor azonosítóval, melynek kód- és adattagjainak azonosítóit és láthatósági szintjét az osztálydiagram szemlélteti! A privát adattagokat egy lakat szimbólum különbözteti meg a publikusaktól. Ékezetes azonosítókat is készíthet, illetve azokat kis- és nagybetűkkel is kezdheti.
- Az EletjatekSzimulátor osztály konstruktora kapja paraméterül a mátrix méretét egész típusú változókból!
- A konstruktor állítsa be a paraméterek értékeivel az adattagok (OszlopokSzama, SorokSzama) értékeit! Ugyancsak a konstruktorban inicializálja a Matrix azonosítójú, egész típusú mátrixot (kétdimenziós tömböt) technikai okokból a megadott méretnél **kettővel több sorral és oszloppal**, majd tölts fel úgy véletlenszerűen 0 vagy 1 értékekkel a „belső cellákat”, hogy a „külső cellák” (szélső sorok és szélső oszlopok) 0 értékűek legyenek mind a négy oldalon!
- Készítse el a megjelenítést végző metódust! A metódus a mátrixot jelenítse meg úgy a konzolon, hogy a „külső cellákat” az „X” karakterrel, a „belső cellákat” 0 érték esetén szóközzel, 1 érték esetén „S” (**sejt**) karakterrel jelölje a minta szerint!
- Kódolja a következő állapotot (kört) meghatározó metódust a következők szerint:
 - A mátrixban lévő **1-es értékeket sejteknek** nevezzük. Egy cella környezete a hozzá legközelebb eső maximum 8 cella (tehát a cellához képest „átlósan” elhelyezkedő cellákat is figyelembe vesszük). Egy sejt vagy üres cella szomszédjai a környezetében lévő sejtek (sejtszomszédok).



```

XXXXXXXXXXXX
XS  S S SSSX
XSSSSSSS SSX
XS  SSSSS  X
X  SS      X
X SSS S    X
XSSSSSSSS X
XS SSSS SS X
X  SS  S SX
XS      S X
X SS SSSSS X
XXXXXXXXXXXX
    
```

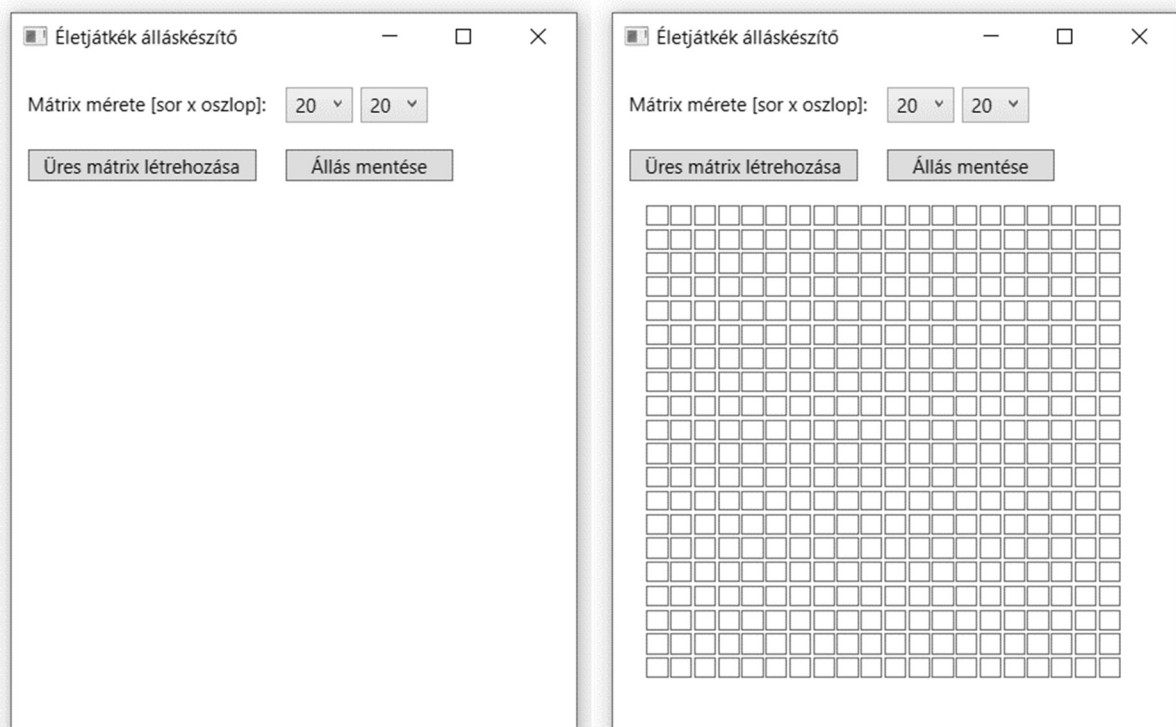
A feladat a következő oldalon folytatódik.

¹ Forrás: <https://hu.wikipedia.org/wiki/Életjáték>

--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

- Egy sejtrel vagy üres cellával egy körben a következő három dolog történhet:
 - A sejt túléli a kört (cella értéke 1 marad), ha két vagy három szomszédja van.
 - A sejt elpusztul (cella értéke 0 lesz), ha kettőnél kevesebb (elszigetelődés) vagy háromnál több (túlnépesedés) szomszédja van.
 - Új sejt születik minden olyan üres (0 értékű) cellában, melynek környezetében pontosan három sejt található.
 - Minden egyes „belső cellára” a sejtszomszédok számának meghatározása után rendre alkalmazza a fenti három szabályt! A sejtszomszédok számának meghatározása az előző állapotú mátrixon történjen, azaz az új állapotot kódoló mátrix értékeit csak a módszer végén töltsse vissza a *Matrix* adattagba!
7. A `Run()` azonosítójú módszerben jelenítse meg a mátrixot, határozza meg a következő kör állapotát a privát módszerek hívásával, majd várakozzon 500 ezredmásodpercet a program végrehajtása!
 8. A főprogramban hozzon létre egy *EletjatekSzimulator* példányt 10x10-es mérettel a minta szerint, majd ismételten hívja a `Run()` módszerét egy billentyű leütéséig!
 9. Készítsen **grafikus alkalmazást**, melynek a projektjét *EletjatekGUI* néven mentse el!

A grafikus alkalmazásban a következő feladatokat végezze el:



- a. Alakítsa ki a felhasználói felületet a fenti, bal oldali minta szerint! Az alkalmazás címsorában megjelenő felirat „Életjáték álláskészítő” legyen! A mátrix méretét a legördülő listák segítségével lehessen kiválasztani, melyek értékeit töltsse fel az alkalmazás indulásakor 5–20 számokkal! Oldja meg, hogy mindkét listában a 20-as érték legyen az alapértelmezett!

A feladat a következő oldalon folytatódik.

--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

- b. Az „Üres mátrix létrehozása” feliratú parancsgomb lenyomása után hozzon létre programjával jelölőnégyzet (CheckBox) példányokat mátrixszerűen elrendezve a kiválasztott méretnek megfelelően a fenti, jobb oldali minta szerint! Ha korábban már volt létrehozott jelölőnégyzet-mátrix a felhasználói felületen, akkor a létrehozás előtt törölje azt!
- c. Az „Állás mentése” parancsgomb lenyomása után hozzon létre szöveges állományt, mely a mátrix sorait kódolja! Az állomány azonosítója például `Eletjatek_5x6.txt` legyen, ahol a számértékek a mátrix méretének megfelelően változnak! Az üres jelölőnégyzeteket 0-val, a kipipált állapotúakat 1-gyel jelölje a következő minta szerint!

Mátrix mérete [sor x oszlop]: 5 6

Üres mátrix létrehozása

Állás mentése

<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>

➔

Eletjatek_5x6.txt

0	0	1	1	0	0
0	1	0	0	1	0
0	1	0	0	1	0
0	1	1	1	1	0
0	1	0	0	1	0