Számítógépes Szimulációk

Sejtautomaták

JEGYZŐKÖNYV

Jakobi Ádám

Tartalomjegyzék

1. Bevezető

A számítógépes szimulációk[1] hetedik, utolsó beadandójában két feladat került kiírásra. Az egyik a Conway-féle életjátékot szimuláló C++ program írása, a másik pedig a 2D-homkdomb modell skálázási szabályának mérése. Célom a modellek megvalósítása és az eredmények ábrázolása, animációk készítése.

2. Conway-féle életjáték

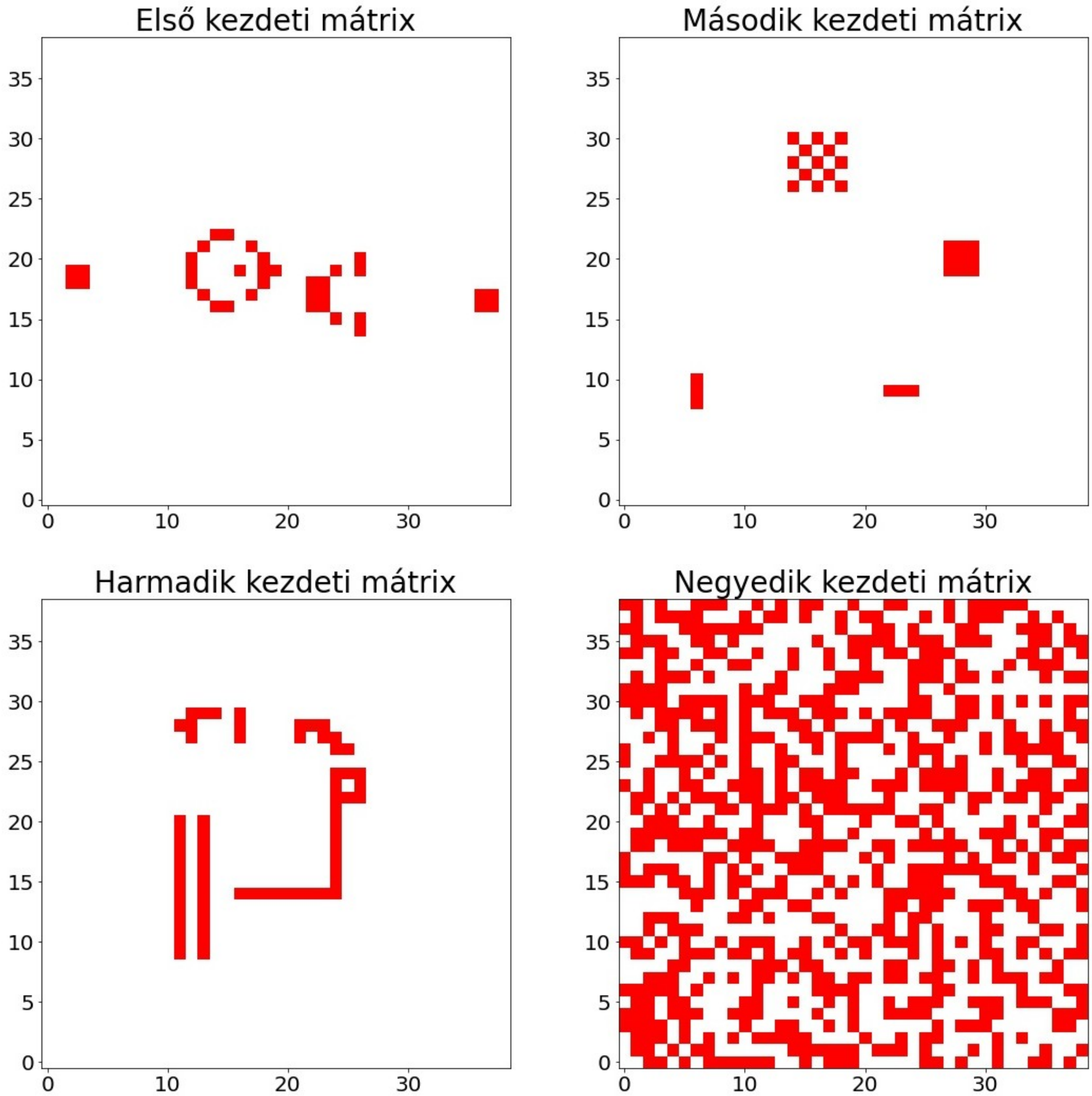
Az életjáték tere felfogható egy mátrixként, ahol minden egyes mátrixelem egy sejt. A mátrixot 0 és 1 egész számokkal kell inicializálni tetszőleges módon. A 0 halott sejtet, az 1 pedig élőt jelent. A mátrix szélei adják a rendszer határait, amit különböző határfeltételekkel választhatunk meg:

- konstans élő sejtek (a keret minden eleme 1),
- konstans halott sejtek (a keret minden eleme 0),
- periodikus határfeltétel,
- véletlen 0 - 1 feltöltés.

A rendszer ezután minden időbeli lépés során az életjáték szabályai szerint frissíti a sejtek állapotát. Ezek a szabályok a következők:

- ha egy sejtnek n élő szomszédja van, akkor a sejt állapota nem változik,
- ha egy sejtnek $n + 1$ élő szomszédja van, akkor a sejt előző állapotától függetlenül élő lesz,
- minden más esetben a sejt halott lesz.

A feladatot a fent említett peremfeltételekre, többféle mátrixfeltöltéssel, $n = 1, 2, 3...8$ esetekben kellett elvégezni. Ehhez minél általánosabb program írása volt célszerű, ahol a perem- és kezdetifeltételek megadhatók a main függvény argumentumaiként. A játék során négy különböző, 40×40 méretű mátrixot használtam, amiket még kibővítettem a határfeltételhez szükséges kerettel. Külön írtam függvényt a mátrix belsejének és a keret elemeinek módosításához. A kezdeti mátrixokat .txt fájlokként tároltam el és olvastam be a program futása során. A kezdeti mátrixok összeállításai tekinthetők meg a következő 1. ábrán.



1. ábra. Kezdeti mátrixok ábrái. Az első mátrixnak a Gosper glider gun elrendezést választottam, a másodikat és a harmadikat tetszés szerint manuálisan hoztam létre, a negyedik elemeit pedig a python random.randint(0,1) függvénnyel töltöttem fel véletlenszerűen.

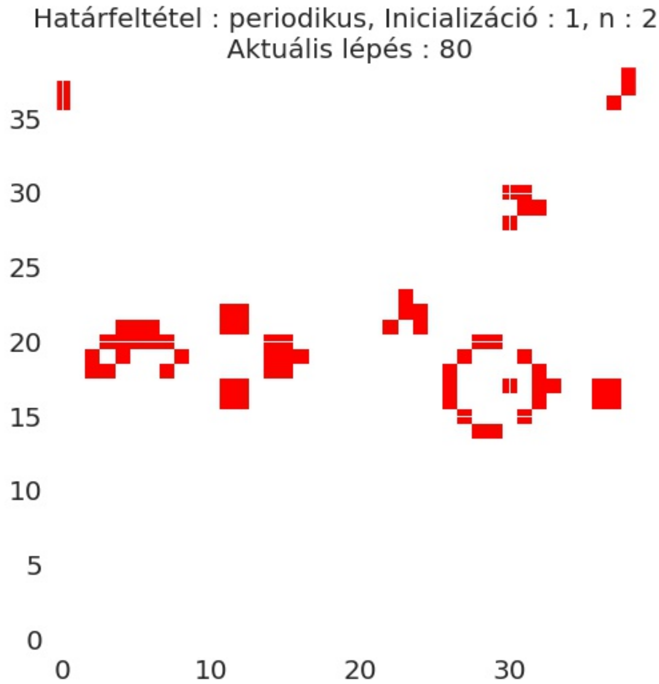
Az eredményeket a megadott paraméterekhez tartozó célmappákba mentettem, minden lépés során egy belső mátrixot írtam ki az adott időbeli lépés indexével. Ha az elemek éppen nem érintkeznek konstans pozitív sejttel a mátrix határán, akkor egy alakzatnak mindig van olyan eleme, aminek három vagy kevesebb szomszédja van. Ez azt jelenti, hogy $n = 4, 5, 6, 7, 8$ esetén bármilyen alakzat, ami nem a keret kihasználásának egy speciális esete (például $n = 7, 8$ esetén, konstans élő kerettel, ha a mátrix minden eleme élő, akkor egy sejt sem fog elpusztulni az idő múlásával) el fog tűnni véges időbeli lépés alatt. $n = 3$ esetén csak olyan állandó alakzatok maradnak fenn, melyek minden elemének legalább három szomszédja van (erre példa a 2×2-es négyzet, vagy az ötelemű plusz jel). $n = 1$-nél a sejtek könnyen elterjedhetnek, így rövid időn belül beborítják az egész táblát, terjedésüknek csak a túlszaporodás vet gátat. Így végeredményben $n = 1$-re úgy néz ki a kép, mint egy régi televízió, amin elment az adás (grízes kép, mászó hangyák) vagy mint egy véletlenszerűen változó QR kód. Ennélfogva $n = 2$ marad az egyetlen olyan eset, amihez igazából érdemes ábrát és animációt készíteni. Én végül az $n = 1, 2, 3$ esetekről készítettem animációkat, hogy a sejtek elhalási folyamatát és az esetlegesen megmaradó konstans alakzatokat is megfigyelhetőkké tegyem, a sejtek kaotikus túlszaporodásával egyetemben. Az animációkat python programozási nyelv segítségével készítettem el, melyek megtekinthetőek a kooplexen, a feladat mappájában (ezek ha túl sok helyet foglalnak, nyugodtan törölhetők). Ezenkívül az $n = 2$ esetek animációit youtube-on[2] is elérhetővé tettem, a lejátszási lista linkje megtalálható a hivatkozások között. A többi ($n \geq 4$) esetnél azt tapasztaltam, hogy általában pár lépés alatt minden belső sejt (tehát ami nem része az esetlegesen konstans élő keretnek) elpusztul, ezért az elméleti magyarázaton kívül nem részletezem bővebben az $n = 4$-nél nagyobb eseteket. A konklúzió, hogy változatos mozgások és érdekes mintázatok $n = 2$ esetén alakulhatnak ki.

3. Diszkusszió

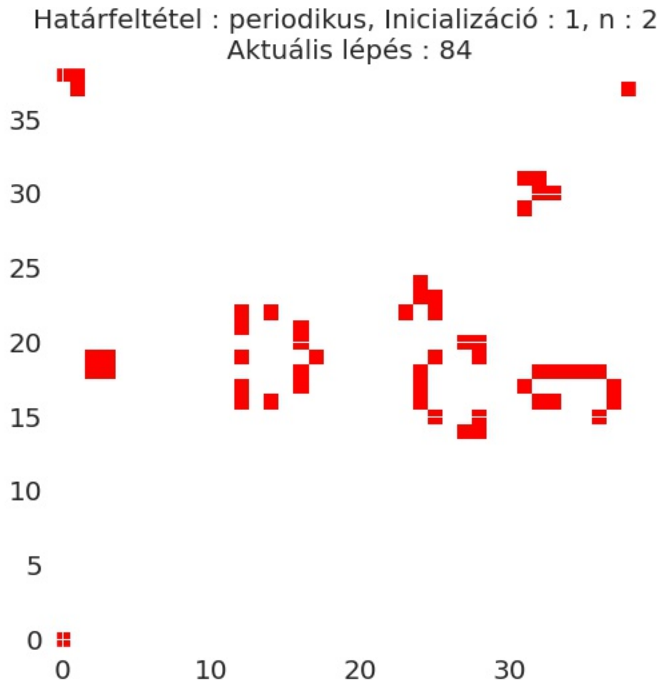
A számítógépes szimulációk hetedik, utolsó beadandójában implementáltam és diszkutáltam a Conway-féle életjáték különböző eseteit, az eredményekről animációkat is készítettem, melyek elérhetők a kooplex oldalán, a feladat mappájában és a fontosabb videókat a youtube-ra is feltöltöttem[2]. A 2D-homokdomb modell skálázási szabályának mérésére nem maradt időm, annak feldolgozását kihagytam.

4. Függelék

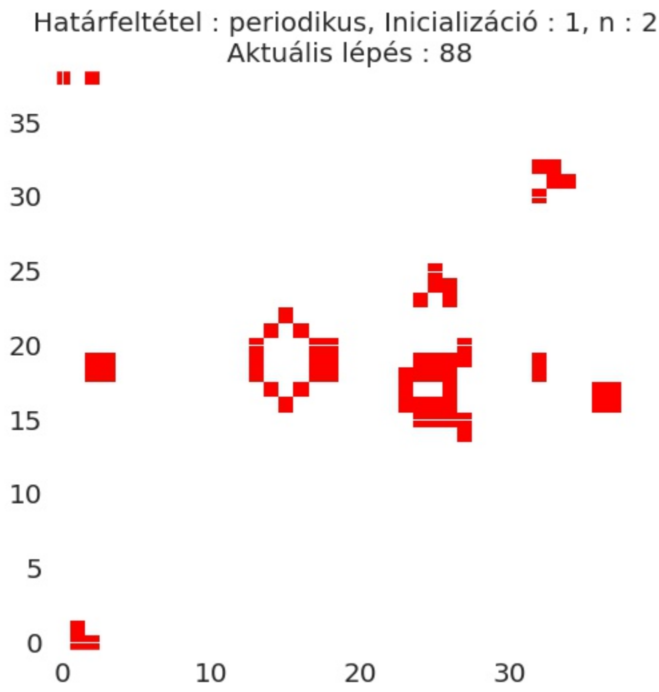
4.1. Pár kép a szimulációk egyes lépéseiről



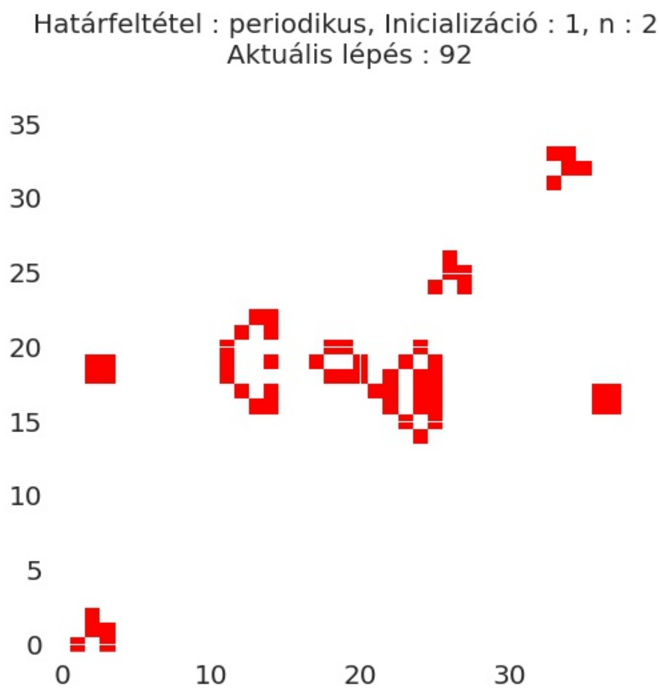
2. ábra



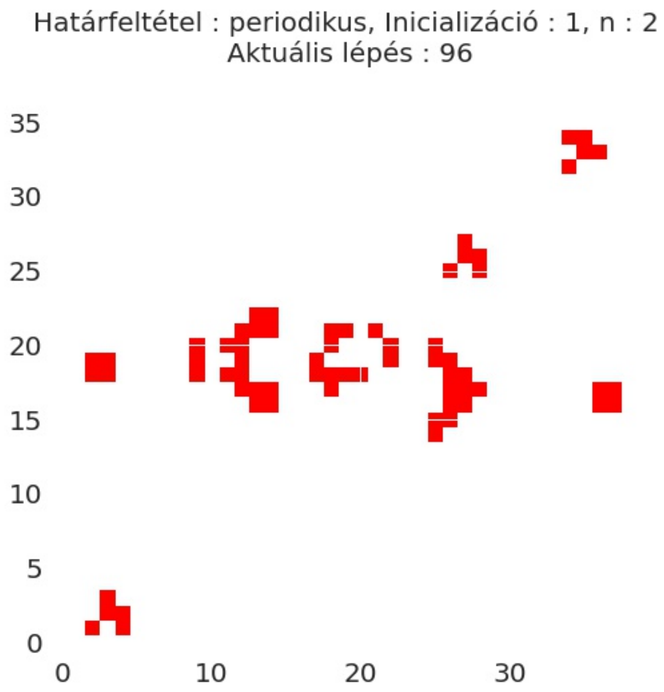
3. ábra


Határfeltétel : periodikus, Inicializáció : 1, n : 2
Aktuális lépés : 88
35
30
25
20
15
10
5
0
0
10
20
30

4. ábra


Határfeltétel : periodikus, Inicializáció : 1, n : 2
Aktuális lépés : 92
35
30
25
20
15
10
5
0
0
10
20
30

5. ábra



6. ábra

Hivatkozások

[1] „Számítógépes szimulációk weboldal.” https://stegerjozsef.web.elte.hu/teaching/szamszim/, 2021.

[2] J. Ádám, „Link az animációk youtube lejátszási listájához.” https://www.youtube.com/playlist?list=PL4gVMn1LI9jONBcQMX3sUwctuNpqOL5oM, 2021.