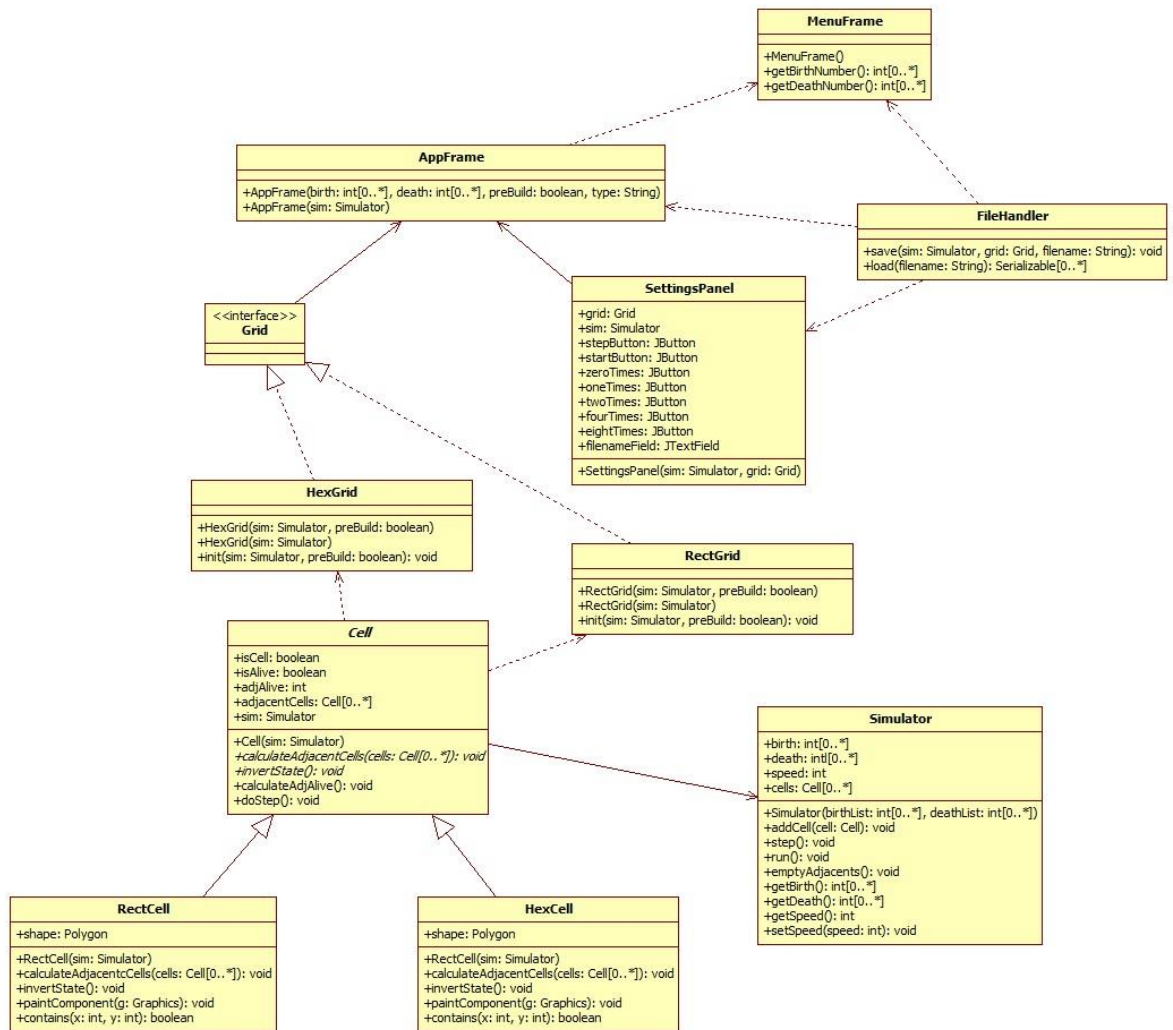


Programozás alapjai 3 – Házi Feladat

Dokumentáció

Arató Bence Bálint (H2TFTH)

I. Osztálydiagram



II. Metódusok leírásai:

1. MenuFrame:

public List<Integer> getBirthNumber(): Listába teszi a születéshez szükséges élő szomszédok számát, érvénytelen érték esetén visszaad egy listát, amibe beleteszi a hármas számot.

public List<Integer> getDeathNumber(): Listába teszi a túléléshez szükséges élő szomszédok számát, érvénytelen érték esetén visszaad egy listát, amibe beleteszi a kettes és a hármas számot.

StartButtonListener.actionPerformed(ActionEvent e): Kiolvassa a paramétereket a komponensekből, majd létrehozza a szimulációt tartalmazó AppFrame-t.

LoadButtonListener.actionPerformed(ActionEvent e): Létrehoz egy FileHandler objektumot és meghívja a load metódusát. Az eredménnyel létrehoz egy új AppFrame-t.

2. Cell:

public abstract void calculateAdjacentCells(): Absztrakt metódus, a szimuláció indításakor egy listába teszi a sejt szomszédjait.

public void calculateAdjAlive(): Végigmegy a sejt szomszédjain, és megszámlálja mennyien élnek közülük.

public void invertState(): Invertálja a sejt állapotát, vagyis léső sejtből halottat, halott sejtből pedig élőt csinál, majd ezután újrifesti a sejtet.

public void doStep(): Lépteti a sejtet. Megvizsgálja kell-e változnia a sejt állapotának a szabályok alapján, és amennyiben igen invertálja azt.

ButtonListener.actionPerformed(ActionEvent e): Akkor hívódik meg, ha a felhasználó a sejtre kattint. Ilyenkor invertálja a sejt állapotát, amennyiben a szimuláció még nem indult el.

3. RectCell:

public void paintComponent(Graphics g): Megfesti a sejtet az éppen aktuális állapota szerint.

public boolean contains(int x, int y): Kattintáskor hívódik meg, visszaadja, hogy a kattintás ezen a sejten történt-e.

4. HexCell:

public void paintComponent(Graphics g): Megfesti a sejtet az éppen aktuális állapota szerint.

public boolean contains(int x, int y): Kattintáskor hívódik meg, visszaadja, hogy a kattintás ezen a sejten történt-e.

5. RectGrid:

public void init(Simulator sim, boolean preBuild): Inicializálja a játékteret, vagyis feltölti Cell objektumokkal. Ha preBuild igaz, akkor invertálja a létrehozott sejtek állapotát, ezzel hozzáadva őket a szimulációhoz.

6. HexGrid:

public void init(Simulator sim, boolean preBuild): Inicializálja a játékteret, vagyis feltölti Cell objektumokkal. Ha preBuild igaz, akkor invertálja a létrehozott sejtek állapotát, ezzel hozzáadva őket a szimulációhoz.

7. Simulator:

public void addCell(Cell cell): Hozzáadja cell sejtet a cells listához.

public void step(): Két ciklust futtat le a cells listán. Az első megszámlolja a sejtek élő szomszédjait, a második ettől függően invertálhatja az állapotukat.
public void calculateAdjacentCells(): A szimuláció indulásakor fut le, egy ciklus meghívja minden sejt calculateAdjacentCells() metódusát.
public void run(): A Thread osztály run metódusának felülírása. A sebességtől függő időközönként meghívja a step() metódust.
public void emptyAdjacents(): Kiüríti minden sejt szomszédait tároló listát, felkészítve a sejtet a sorosított mentésre.

8. FileHandler:

public void save(Simulator sim, Grid grid, String filename): Sorosítással elmenti sim és grid objektumokat filename nevű fájlba.
public void load(String filename): Betölti a sorosított Simulator és Grid objektumokat, majd visszaadja azokat egy listában. Hiba esetén null értékkel tér vissza.

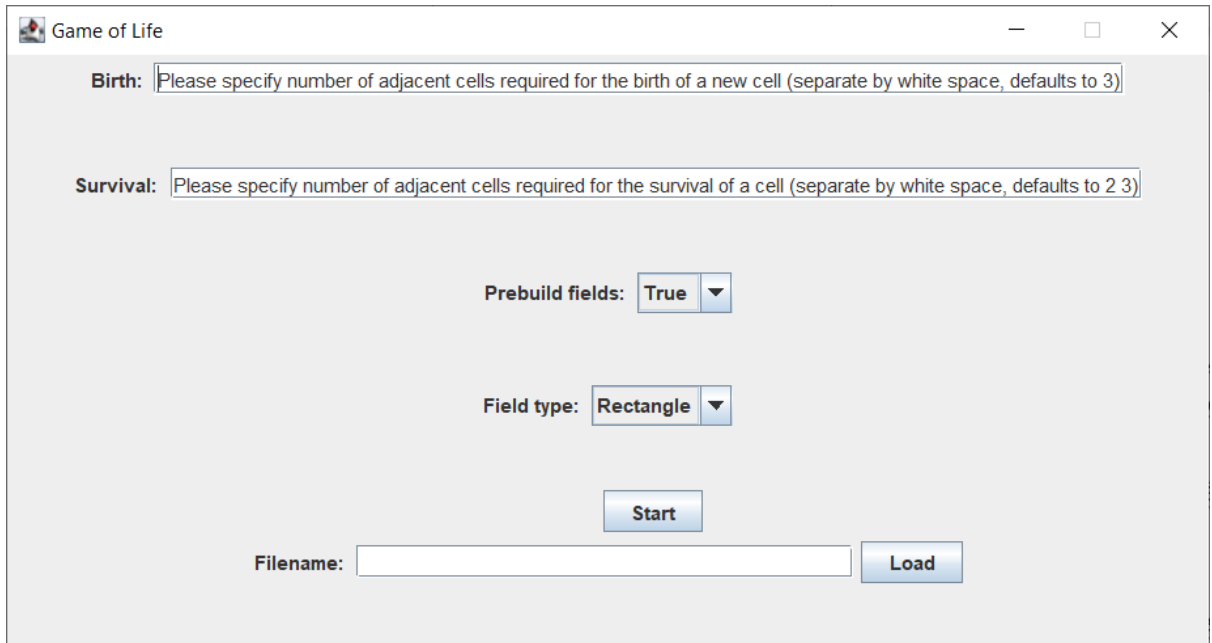
9. SettingsPanel:

TimeButtonListener.actionPerformed(ActionEvent e): Ha a szimuláció elkezdődött átállítja annak sebességét az osztály konstruktorában kapott értékre.
StartButtonListener.actionPerformed(ActionEvent e): Elindítja a szimulációt, vagyis kiszámolja sejtek szomszédjait és átállítja a szimuláció sebességét nullára.
StepButtonListener.actionPerformed(ActionEvent e): Amennyiben a szimuláció sebessége nulla, lépteti egyet a szimulációt.
SaveButtonListener.actionPerformed(ActionEvent e): Létrehoz egy FileHandler objektumot és meghívja a save metódusát.

III. Felhasználói kézikönyv

Menü

Amikor a megnyitja az alkalmazást a menu fogadja. Itt tudja beállítani a szimuláció paramétereit, illetve betölteni korábbi szimulációkat.



The screenshot shows a window titled "Game of Life" with standard Windows window controls (minimize, maximize, close). Inside the window, there are several input fields and buttons:

- Birth:** A text input field with placeholder text: "Please specify number of adjacent cells required for the birth of a new cell (separate by white space, defaults to 3)".
- Survival:** A text input field with placeholder text: "Please specify number of adjacent cells required for the survival of a cell (separate by white space, defaults to 2 3)".
- Prebuild fields:** A dropdown menu currently showing "True".
- Field type:** A dropdown menu currently showing "Rectangle".
- Start:** A button located below the dropdowns.
- Filename:** A text input field.
- Load:** A button located to the right of the filename field.

A *"Birth"* szövegmező kitöltésével adjhatja meg, hogy mennyi élő szomszédal kell a sejteknek rendelkeznie, hogy maguk is életre keljenek. Ha több számot kíván megadni, azokat szóközzel válassza el. Amennyiben üresen hagyja, vagy érvénytelen bemenetet ad meg, akkor a program automatikusan a hármas számmal dolgozik tovább.

A *"Survival"* szövegmező kitöltésével adjhatja meg, hogy mennyi élő szomszédal kell az élő sejteknek rendelkeznie, ahhoz hogy életben maradjanak. Ha több számot kíván megadni, azokat szóközzel válassza el. Amennyiben üresen hagyja, vagy érvénytelen bemenetet ad meg, akkor a program automatikusan a kettes és a hármas számmal dolgozik tovább.

A *"Prebuild fields"* combobox-ban választhatja ki, hogy maga szeretné a játékteret felépíteni, vagy egy előre felépített játéktérben futtatná a szimulációt. Ha a *"True"* értéket választja egy kitöltött pályán indul meg a szimuláció, *"False"* érték esetén a maga építheti meg a játékteret egy üres táblán.

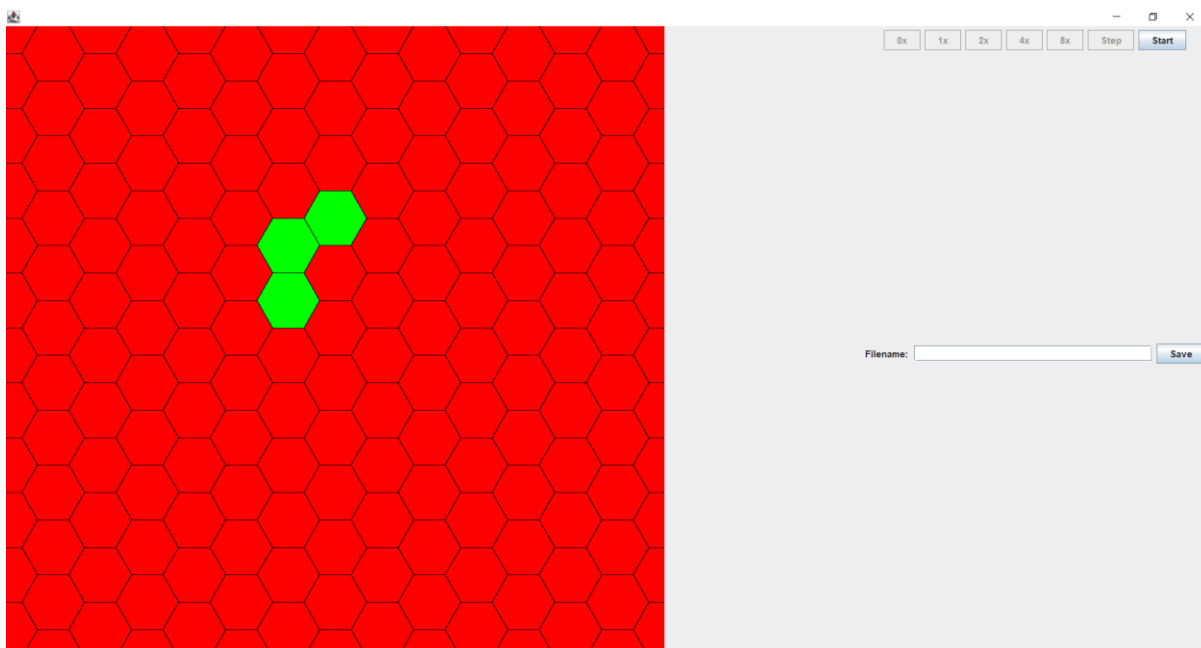
A *"Field type"* combobox-ban választhatja ki a mezők típusát. Válassza a *"Rectangle"* értéket négyzet alakú mezőkhöz, illetve a *"Hexagon"* értéket hatszög alakú mezőkhöz.

A paraméterek megadása után a *"Start"* gomb megnyomásával indíthatja el a szimulációt.

A *"Filename"* szövegdobozba beírhatja egy mentett szimuláció nevét, amelyet a *"Load"* gomb megnyomásával tölthet be.

Szimuláció

Miután megnyomja a *“Start”* gombot, megjelenik a szimuláció egy új ablakban.



Amennyiben a *“Prebuild”* combobox-ban a *“True”* értéket választotta a program legenerálja a szimuláció játéktérét. A játéktér piros színnel jelzi, ha egy sejt halott, illetve zöld színnel ha él. A szimuláció indításáig a sejtek kattintással hozhatóak létre, és állapotuk is kattintással változtatható meg

A szimuláció a *“Start”* gomb megnyomásával indítható el, a szimuláció sebessége ilyenkor nulla. A sebességet a *“0x”*, *“1x”*, *“2x”*, *“4x”*, *“8x”* gombok megnyomásával tudja irányítani.

A szimuláció egyesével is léptethető. Ehhez először állítsa meg a szimuláció a *“0x”* gomb megnyomásával. Ezután a *“Step”* gombra kattintva léptetheti a szimulációt.

Amennyiben el szeretné menteni a szimulációt először állítsa le azt a *“0x”* gombbal. Ezután a *“Filename”* szövegmezőben adja meg milyen néven kívánja menteni a szimulációt, majd kattintson a *“Save”* gombra.