**Tárgy: Programozási technológiák**

**tárgykód: IP—18cPROGTEG 11**

**Gyakorlatvezető: Novák Márk Attila**

**Készítette: Böröczky Bálint Patrik**

**Neptun kód: j7q688**

**Feladatsorszám: 4**

Tartalom

[1.Feladat szövege : 3](#_Toc118732627)

[2.Feladat elemzése : 4](#_Toc118732628)

[3.UML diagram 4](#_Toc118732629)

[Field osztály : 5](#_Toc118732630)

[Board osztály : 5](#_Toc118732631)

[Board konstruktor : 5](#_Toc118732632)

[isOver : 5](#_Toc118732633)

[BoardGUI a feladat megoldása osztály 5](#_Toc118732634)

[BoardGUI konstruktor : 5](#_Toc118732635)

[elapsedTime : 5](#_Toc118732636)

[delete3 : 5](#_Toc118732637)

[delete4: 5](#_Toc118732638)

[keres1: 5](#_Toc118732639)

[keres2: 6](#_Toc118732640)

[refresh : 6](#_Toc118732641)

[mennyi : 6](#_Toc118732642)

[torlesReal1elem 6](#_Toc118732643)

[checkViz 6](#_Toc118732644)

[checkFug 6](#_Toc118732645)

[checkKeresztPos 6](#_Toc118732646)

[checkKeresztNeg 6](#_Toc118732647)

[AmobaGUI a pálya kialakító (+funkciók) 6](#_Toc118732648)

[AmobaGUI konstruktor : 6](#_Toc118732649)

[4.Esemény – eseménykezelés 7](#_Toc118732650)

[Esemény : kattintás (refresh + buttonListener) 7](#_Toc118732651)

[Esemény : új játék kezdése (game-> pályaválasztás {AmobaGUI actionPerformed) 7](#_Toc118732652)

[Esemény : exit ->kilépés a játékból (AmobaGUI actionPerformed exit) 7](#_Toc118732653)

[Esemény : győztes -> Player1 vagy Player2 (BoardGUI buttonListener) 7](#_Toc118732654)

[Esemény : nincs nyertes,betelik a tábla (BoardGUI buttonListener) 8](#_Toc118732655)

[Esemény : játékos változás 8](#_Toc118732656)

[5.Továbbfejlesztési lehetőségek: 9](#_Toc118732657)

[6.Program müködése : 9](#_Toc118732658)

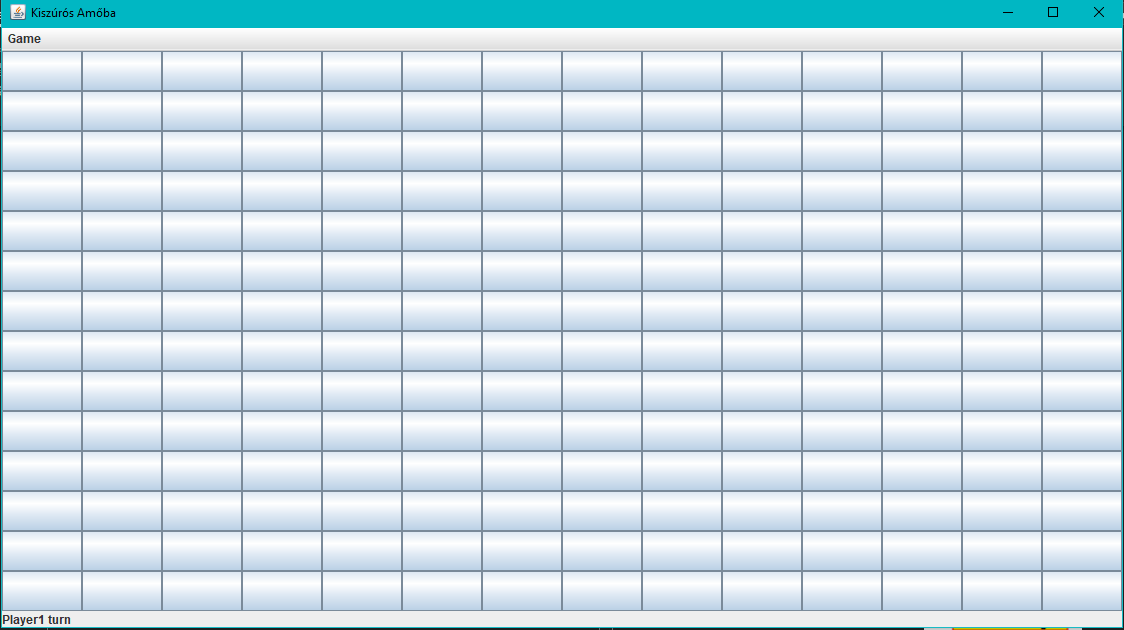
# 1.Feladat szövege :

**Kiszúrós amőba**

Készítsünk programot, amellyel a közismert amőba játék következő változatát játszhatjuk. Adott egy n × n-es tábla, amelyen a két játékos felváltva **X**, illetve **O** jeleket helyez el. Csak olyan mezőre tehetünk jelet, amely még üres. A játék akkor ér véget, ha betelik a tábla (döntetlen), vagy valamelyik játékos kirak **5** egymással szomszédos jelet **vízszintesen**, **függőlegesen** vagy **átlósan**. A program minden lépésnél jelezze, hogy melyik játékos következik, és a tábla egy üres mezőjét kijelölve helyezhessük el a megfelelő jelet.

A kiszúrás a játékban az, hogy ha egy játékos eléri a **3** egymással szomszédos jelet, akkor a program automatikusan törli egy jelét egy véletlenszerűen kiválasztott pozícióról (nem biztos, hogy a hármasból), ha pedig **4** egymással szomszédos jelet ér el, akkor pedig kettőt.

A program biztosítson lehetőséget új játék kezdésére a táblaméret megadásával **(6×6, 10×10, 14×14),** és ismerje fel, ha vége a játéknak. Ekkor jelenítse meg, hogy melyik játékos győzött (ha nem lett döntetlen), majd kezdjen automatikusan új játékot.



# 2.Feladat elemzése :

-Amőba játék elkészítése különböző nagyságú pályák választásával.

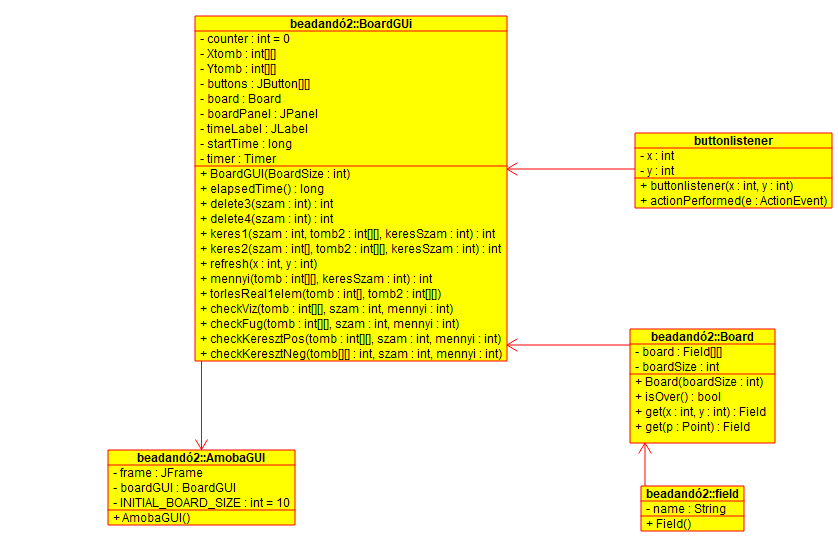
-Egyes esetek esetén a „kiszúrás” történik.2 féle mindez

- Ha van három szomszédos -> akkor egy véletlen eltűnik a jelöltjeink közül(ez lehet a szomszédosok között is)

-Ha van négy szomszédos -> ekkor kettő véletlen eltűnik a jelöltjeink közül(szintén lehet szomszédosok között){megjegyzés : mivel ha van négy szomszédos az garantálja azt is hogy van három is}

-Ha sikerül kitenni mindezek ellenére 5 szomszédost akkor az adott játékos nyer és automatikusan újra indul a játék (10x10-es mérettel).

# 3.UML diagram



## Field osztály :

A képen szöveg látható

Automatikusan generált leírás-A field osztály képezi a játékunk mezőit.

-A konstruktorában a nevet null-ra állítjuk.

## Board osztály :

### A képen szöveg látható Automatikusan generált leírásBoard konstruktor :

A konstruktorban kapott méret alapján feltölti

a board-ot fieldekkel.

### isOver :

Ellenőrzi , hogy van-e szabad mező ha nincs false-al tér vissza míg ha nincs akkor vége a játéknak és true-t ad vissza.

## A képen szöveg látható Automatikusan generált leírásBoardGUI a feladat megoldása osztály

### BoardGUI konstruktor :

Feltölti a pályát „gombokkal” melyekre kattintva el tudunk helyezni x-et és o-t.

Beállítja a gombok méretét.

Illetve kiírja bal alul ki jön.

### elapsedTime :

Eltelt időt számolja ki.

### delete3 :

3 szomszédos esetén hívódik meg. Egy véletlen számmal tér vissza 0 és szam között.(ahol a szám = x vagy o -ök számával)

### delete4:

4 szomszédos esetén hívódik meg. Két különböző számmal tér vissza.(szam = x vagy y-ok száma)

### keres1:

3 szomszédos esetén használandó. Megkeresi azt amit törölni kell az adott tömbből (Lehet ez X vagy az O tomb). A keresSzam = 1 vagy 2 (attól függően melyik tömbben keresünk 1 – X 2 – O).

### keres2:

4 szomszédos esetén használandó. Megkeresi azokat amiket törölni kell.(törlendő számok : szam tomben) . A keresSzam = 1 vagy 2 (attól függően melyik tömbben keresünk.)

### refresh :

Frissiti a boardot -> kattintás számától függően kerül x vagy o a táblára.Illetve minden kattintás után csekkolja hogy hány szomszéd van kattintás után.

### mennyi :

Visszaadja mennyi x vagy o van elhelyezve a táblán.

A képen szöveg látható

Automatikusan generált leírás

### torlesReal1elem

Az adott értéknek megfelelő helyen visszaállítja a mezőt null-ra -> ezáltal újra elhelyezhető rajta x vagy o.

### checkViz

Vízszintesen ellenőrzi, hogy a paraméterben átadott számnak megfelelő szomszéd van-e.

### checkFug

Függőlegesen ellenőrzi, hogy a paraméterben átadott számnak megfelelő szomszéd van-e.

### checkKeresztPos

Átlósan ellenőrzi, hogy a paraméterben átadott számnak megfelelő szomszéd van-e.(Pos-> monoton csökkenő átló)

### checkKeresztNeg

Átlósan ellenőrzi, hogy a paraméterben átadott számnak megfelelő szomszéd van-e.(Neg-> monoton növekvő átló)

## AmobaGUI a pálya kialakító (+funkciók)

A képen szöveg látható

Automatikusan generált leírás

### AmobaGUI konstruktor :

Létrehozza kezdetben az INITIAL\_BOARD\_SIZE méretű pályát futtatáskor.

Funkciókat ad a panelhez:

* játékméret változtatás (6,10,14)
* illetve az exit funkciót

## 4.Esemény – eseménykezelés

## Esemény : kattintás (refresh + buttonListener)

-Attól függően melyik játékos van soron

-Player1 : X kerül a -Player2 : O kerül a tábla adott helyére ha üres.

-Kattintás után ellenőrzés fut le -> kell-e törölni?

A képen szöveg látható

Automatikusan generált leírás

## Esemény : új játék kezdése (game-> pályaválasztás {AmobaGUI actionPerformed)

-A játék a kiválasztott pályamérettel újrakezdődik.

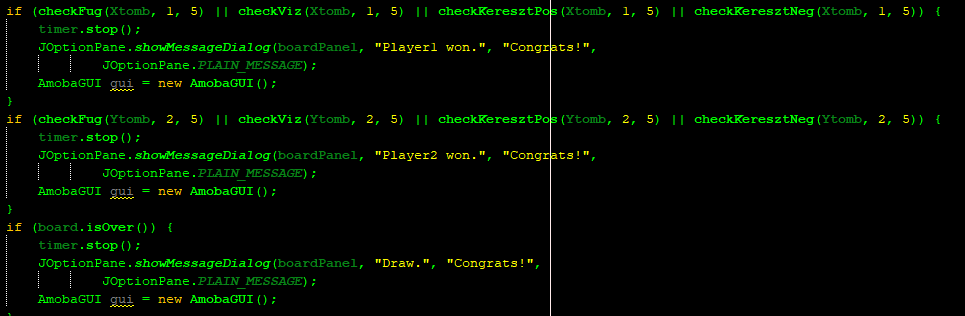
## Esemény : exit ->kilépés a játékból (AmobaGUI actionPerformed exit)

-A játék bezárul->a program futása lezárul.

## Esemény : győztes -> Player1 vagy Player2 (BoardGUI buttonListener)

-A játék nyertesét kiírja majd automatikusan új játékot kezd az INITIAL\_BOARD\_SIZE – zal. Mindez a Amoba GUI „újraindításával”

## Esemény : nincs nyertes,betelik a tábla (BoardGUI buttonListener)

-A végeredmény egy döntetlen majd automatikusan új játékot kezd az INITIAL\_BOARD\_SIZE – zal.->Mindez a Amoba GUI „újraindításával”

## A képen szöveg látható Automatikusan generált leírásEsemény : játékos változás

-Bal alsó sarokban kiírja a soron lévő játékost.

A képen asztal látható

Automatikusan generált leírásA képen asztal látható

Automatikusan generált leírás Player1 soron Player2 soron

# 5.Továbbfejlesztési lehetőségek:

1. Név megadása
2. AI hozzáadása
3. Elmenthető játék

# 6.Program müködése :

1. A képen asztal látható

   Automatikusan generált leírásElindítjuk :
2. A képen asztal látható

   Automatikusan generált leírásPlayer1 rak
3. Player2 rak

A képen asztal látható

Automatikusan generált leírás

1. A képen asztal látható

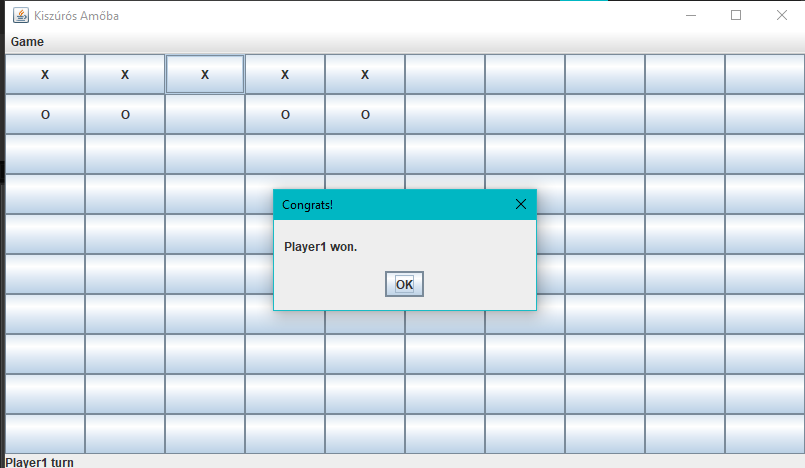
   Automatikusan generált leírásPlayer1-nek megvan a három-> ezért 1 törlődik. (kép csak illusztráció a törlés azonnali)
2. A képen szöveg, szekrény látható

   Automatikusan generált leírásA képen szöveg, szekrény, képernyőkép látható

   Automatikusan generált leírásA képen szöveg, szekrény látható

   Automatikusan generált leírás5.Player1 -nek meglenne 4 de töröl 2-t.

6.Player1 nyer



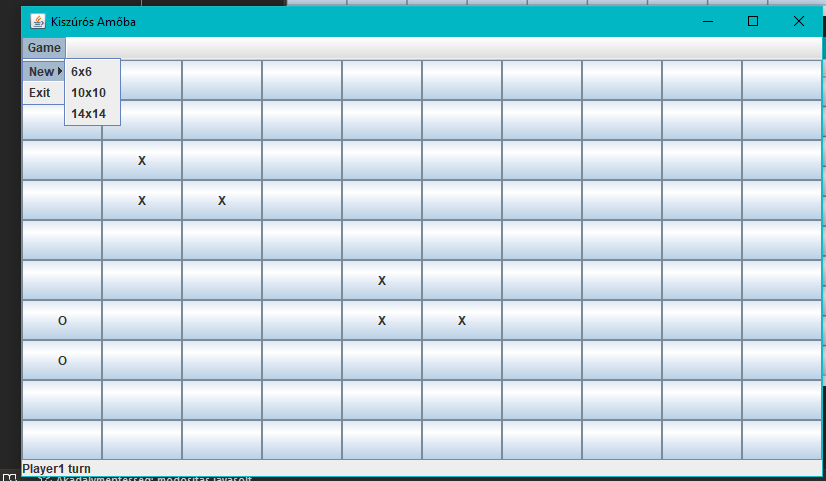
7.Új játé indul

A képen asztal látható

Automatikusan generált leírás

Player2 esetén is ugyanezek történnek.

Új játék indítása :



Kilépés a programból :

