

Specifikáció

A feladatom egy **rendzsu** játék megvalósítása, ami az amőba egy igazságosabb változata. A rendzsut 15×15-ös gó-táblán 2 játékos játssza. Az egyik játékos a fekete, míg a másik a fehér kövekkel rendelkezik. A játékosoknak felváltva kell egyet-egyet lépniük, és öt követ kell egymás mellé helyezni a győzelem elérésének érdekében (vízszintesen, függőlegesen vagy átlósan, megszakítás nélkül), azzal a kitéttel, hogy az első játékosra (fekete) szigorúbb szabályok vonatkoznak:

- nem csinálhat dupla hármast
- nem csinálhat dupla négyest
- nem csinálhat több, mint ötöst

Ha ilyet csinál a fekete (olyan összefüggő ponthalmazt, amely a sík két irányában is tartalmaz 3, illetve 4 hosszú sort), akkor veszít.

Ezen kívül a játék elején egy speciális szekvencia segítségével teszik le az első pár követ, biztosítva, hogy egyenlő esélyekkel induljanak.

Nyitás

- A fekete egymás után három lépést játszik (két fekete, és egy fehér követ rak le).
- Ezután a fehér vagy cserél a feketével, vagy mindketten megtartják eredeti pozíciójukat. Ezen később már nem lehet változtatni.
- A negyedik lépésben a fehér lerakja egy követ.
- A feketének kétféle javaslatot kell felvetnie az ötödik lépést illetően, melyekből aztán a fehér kiválasztja a neki megfelelőt.

A játék megnyerése

Az a játékos nyer, aki először rak ki öt követ egy töretlen sorba. A fehér esetében az overline (ötnél hosszabb sor) is győzelemnek számít.

Ha a feketének összefüggő dupla hármasa, dubla négyese vagy overline-ja keletkezik, akkor a fehér győz.

Döntetlen akkor van, ha a táblán már nem maradt szabad hely. A program magától megállapítja, ha a játéknak vége, ekkor kiírja a kimenetel eredményét.

Ekkor több kő már nem rakható le.

Use-case-ek

A felhasználó a programban tud játszani, valamint pályát kimenteni (tetszőleges helyre és névvel) és betölteni.

A funkciókat (pl kimentés, stb.) a felhasználó gombokkal és a menüsorból érheti el, visszajelzést JLabelen kap.

A program játék közben kiírja, hogy ki lép. A pályán való kattintással lehet követ elhelyezni.

Dokumentáció

User manual

A program egy egyablakos játék, melynek baloldalán a pálya helyezkedik el, jobbról mellette a játék újrakezdésére, a játék kimentésére és betöltésére szolgáló gombok. A kimentés és betöltés funkciót elérhetjük a menüsorból is. A kimentés és betöltés mind befejezett, mind befejezetlen játéknál megtehető, de csak a kezdeti szekvencia végezte után (ilyen korai szakaszban nem is lenne sok értelme elmenteni egy játékot), ezeknél Jchoosert is alkalmaztam.

A játékot mindig a fekete kezdi, majd a színek felváltva következnek(kivéve a nyitás során). A játékosok a pályán való kattintással helyeznek el követ. Olyan pontra kattintva, amin már van kő, másik kő már nem helyezhető el. A játékot „színesíti”, hogy a nyitás során, az 5.lépésben a fekete által felajánlott pontok kezdetben piros színűek.

Osztályok

Board

A Board osztály egy kétdimenziós, inteket tartalmazó tömböt, és egy színnel rendelkező pontokat tároló ArrayListet, valamint a színek jelölésére szolgáló statikus adattagokat foglalja magába. A kétdimenziós tömb jelképezi a táblát, a lista pedig az egyes lépések által meghatározott színes pontokat tartalmazza. A Boardnak konstruktoron kívül a tömb és a lista elérését szolgáló publikus metódusai vannak.

ColorPoint

A színes pontokat reprezentálja (egy Point és egy int(szín) adattagja van), ilyeneket tárol a Board osztály ArrayListje.

Player

A játékos osztály. Int adattagja van, ami az aktuális színét jelenti.

Save

A Save osztály a kimentéshez használt metódust tartalmazza, amely szerializálást használ. txt és dat fájlal is működik.

Load

A Load osztály a beolvasást szolgáló metódust tartalmazza. txt és dat fájlal is működik.

Main

A Main osztályból indul a futtatás. A main metódus példányosít egy Window objektumot.

Window

A játékot tartalmazó ablakosztály. A főpaneljén két kisebb panel található: A BoardPanel, amely a pályát tartalmazza, valamint a rightPanel, amelyen gombok vannak. Tartalmaz továbbá egy menüsört, ahonnan szintén elérhető a Save és Load funkció. A Window osztályban találhatóak a gombok listenerei is.

BoardPanel

A BoardPanel felelős a pálya és a pontok alacsony szintű grafikai kirajzolásáért, a Game-et példányosítja, valamint ez tartalmazza a listenert, ami figyeli a pályán való kattintást,és a listához adja a kattintott pontot.

Game

A Game osztály a Board-ot példányosítja, a játék állását meghatározó változókat tartalmaz, valamint ez felelős a számításokért. A setPoints metódus rakja a pontlista elemeit a mátrixba (a Board osztály setColor metódusát használva), továbbá meghívja a count, rule és evaluate metódusokat.Ez a 3 metódus a felelős a játékszabályok helyes alkalmazááért. A count megkapja a pontlista egy elemények koordinátáit és a szint, majd a mátrixban a pontból a sák 8 irányába maximum 5 hosszan „szétfut” és a

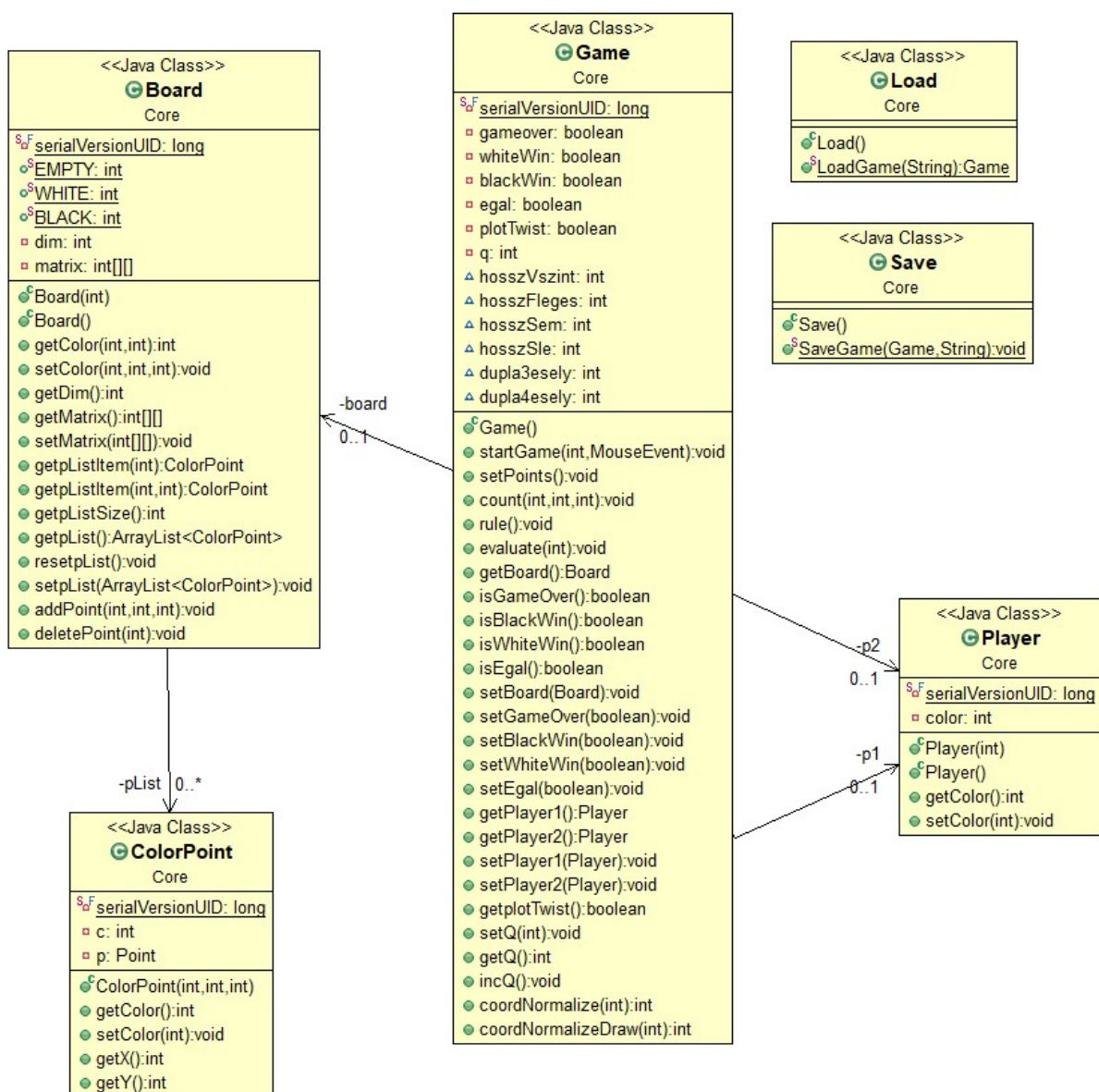
változóban rögzíti, hogy az egyes irányokban a szín kövei milyen hosszú összefüggő sorokat alkotnak. A rule metódus ez előző változók alapján megállapítja, hogy hány darab 3, illetve 4 hosszú sora van a színnek különböző irányokban, ezeket újabb változóban rögzíti. Az evaluate metódus ezek után, a szín tudatában csak a kiértékeli az előző változók értékeit, és kiderül, hogy nyert-e valaki A kimenetel eredményétt ki is írja a Jlabelen.

A Game osztály startgame metódusában próbáltam megvalósítani a játék eleji szekvenciát, mert a meglévő kódot már nem szerettem volna nagyon összerombítani, emiatt viszont a startGame lett elég eklektikus. Változtatásokra természetesen így is szükség volt, főként a BoardPanel osztályban. A starGame metódus lépésként változtatja a szint és a játékost a nyitás leírásának megfelelően, valamint párbeszédablakokat hív a döntéshelyzetekben (legyen-e csere, melyik pontot választja).

A Game osztály többi metódusa az adatok elérését szolgálja.

Az osztálydiagramot az Eclipse beépülőjével készítettem (ObjectAid):

Core:



Gui:

