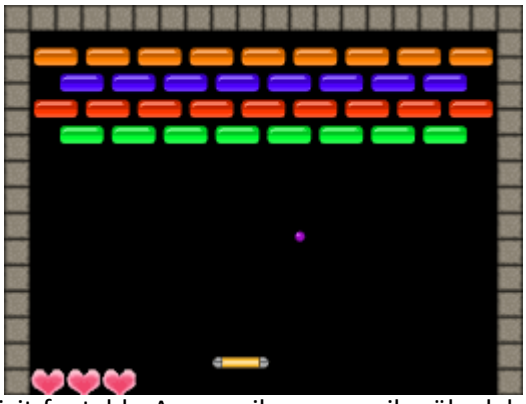
BREAK BRICKS

(magyarul: Téglaromboló)

Név: Vizi Előd

Neptun: S53O1S

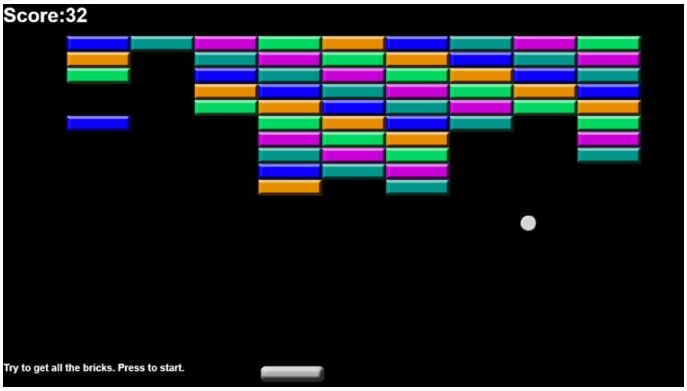
A játék rövid szöveges ismertetése



Ez a játék manapság nagyon elterjedt mobil eszközökön. Alapvetően egy nem túl bonyolult koncepcióra épül: van egy mozgatható elemünk (funkciója minta ping-pong –ban az ütőnek), van egy labdánk, valamint vannak a téglák. Ha a labda nekipattan egy téglának akkor azt lerombolja, visszapattanva róla. Mi ezt a visszapattant labdát kell visszaüssük úgy, hogy lehetőleg sikerüljön egy következő téglát is lebontanunk vele. A téglák alapvetően az alkalmazási ablak felső részében vannak, míg a mozgatható elemünk az ablak aljától picit fentebb. Amennyiben nem sikerül a labdát eltaláljuk, a labda leesik, és veszítünk egy életet. Egy játékos 3 élettel indul és általában több szintből áll ez a játék. Például az első szinten van 10 tégla, a 2. szinten már 15 és így tovább. A tégláknak az elrendezkedése is különböző lehet. Továbbá még egy fontos dolog: a labda az alkalmazási ablak oldalának illetve tetejének ütközve visszapattan.

Ennek a játéknak alapvetően már rengeteg változata megtalálható a különböző telefonos alkalmazás-üzletekben. Van ahol különböző képességeket adó item eshet ki a téglákból, amit ha elkapunk az ütőnkkel, akkor például megnövekszik az, vagy belép még egy labda a játékba, esetleg csökken a gyorsasága a labdának, így könnyebben vissza tudjuk azt ütni stb… Illetve van ahol pályánként nem csak a téglák mennyisége és formációja változik, hanem színmegkülönböztetéssel bővül a repertoár újabb téglákkal, amelyeket esetenként kétszer, háromszor vagy többször is el kell találni, hogy lerombolódjanak. Valamint kerülhetnek be olyan passzív elemek is, amiről tovább pattan a labdánk, de sem lerombolni nem tudjuk sem pedig nem is szükséges lerombolni azt az elemet.

Specifikálás illetve a felhasználói funkciók ismertetése



Az én elképzelésem alapvetően egy a képen látható megvalósítását kivitelezni ennek a játéknak. Ahol a különböző színek határozzák meg, hogy egy téglát hányszor kell eltalálni, hogy azt leromboljuk. Például ha a kék téglát háromszor, a zöldet kétszer és a rózsaszínt egyszer kell eltalálni, hogy eltűnjön. Akkor ha a kék téglát eltaláltuk egyszer átvált a szín zöldre, majd ha megint eltaláltuk ugyan azt, átvált rózsaszínre, legvégül pedig eltűnik. Az én játékomban a játékos 3 élettel fog indulni.

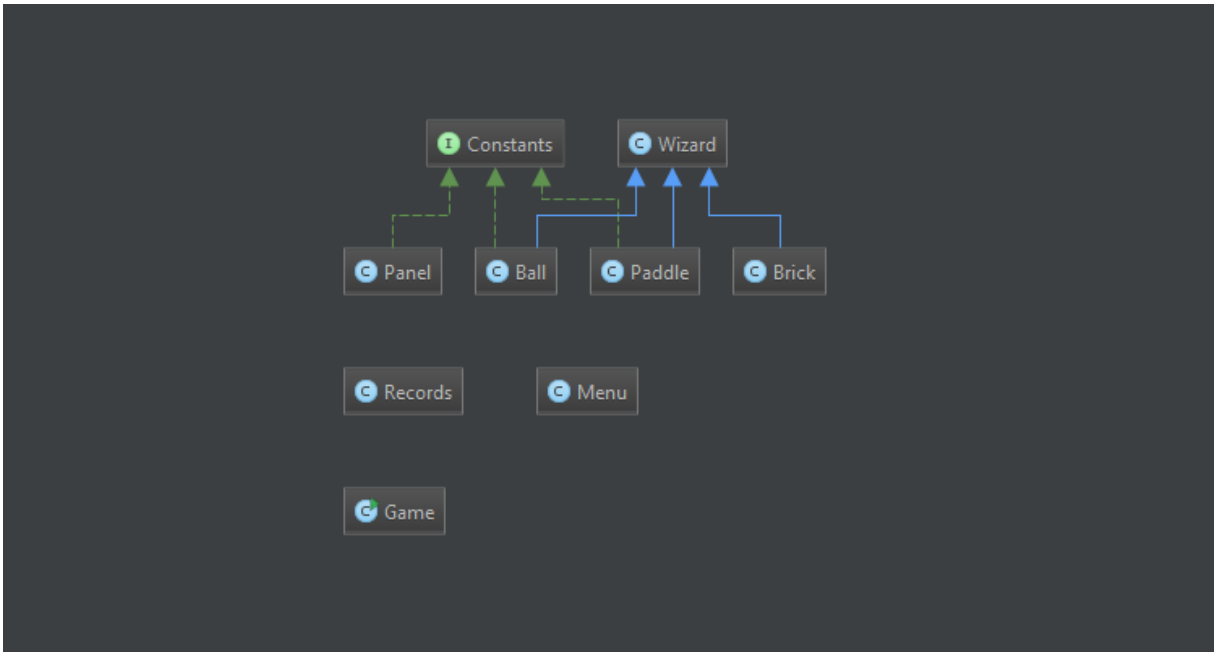
Azt még pontosan nem döntöttem el, hogy alapvetően az időt mérjem, hogy melyik játékos mennyi idő alatt teljesíti a pályát és az alapján legyen egy rangsor, vagy készítek mondjuk 10 szintet, és akkor ki hányadik szintig jut el azzal a 3 életével az számítana. De mindenképp less valamilyen ranglista megvalósítva. Továbbá tervezek egy olyan menüpontot beépíteni, ahol a játékos nehézségi szintet választhat, és ennek a függvényében gyorsabban illetve lassabban mozog majd a labda. Úgyszintén a menüből lesz elérhető a ranglista megtekintése, továbbá a játék indítása. Tehát alapvetően egy start, difficulty, illetve egy ranking menüponton kívül nem tervezek többet beépíteni a játékomba.

A játékos a játék végén adhatja meg a nevét, amikor is a játék kiírja az általa elért pontszámot vagy időt.

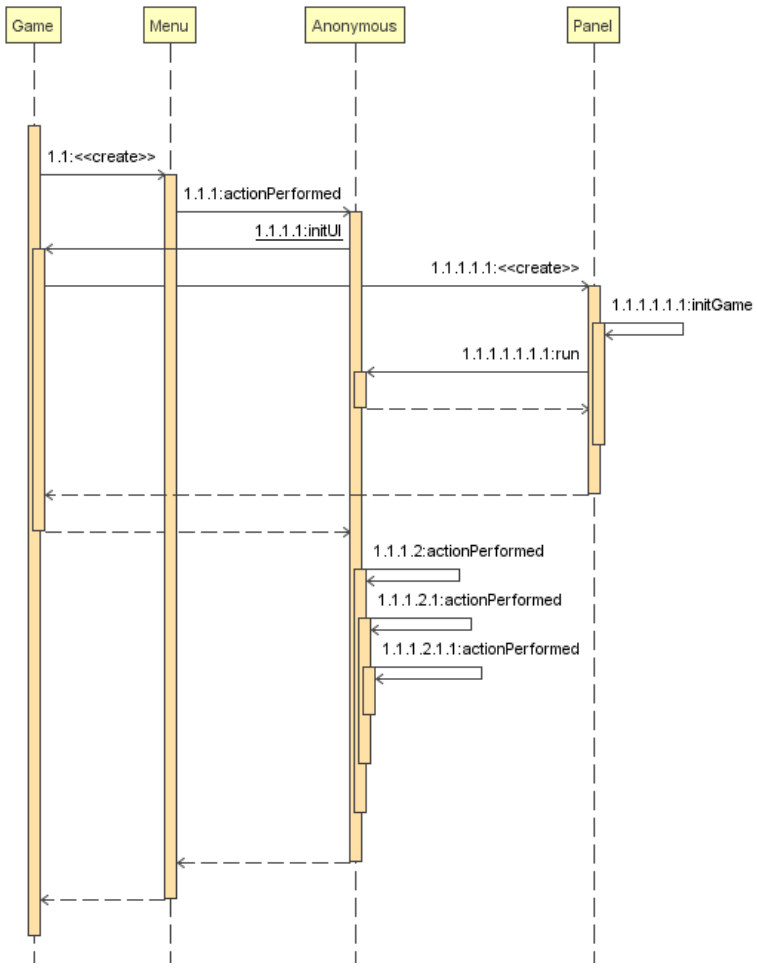
Kivitelezés vázlata

- Swing használata az interfészhez;
- Várhatólag min. 3 osztály használata: labda, ütő, valamint a téglák. de valószínűleg a ranglista kezelésére is külön osztály lesz majd használatos;
- Az ütő mozgatása a billentyűzeten található jobbra illetve balra irányú nyilak használatával lesz megvalósítva;
- Az addigi ranglista (top 10) tárolása egy txt fájlban. Ha szükség van a megjelenítésre, akkor onnan ovlassuk ki, ha változik, akkor abba írunk bele, cask a legjobb tízet tárolja, Ha valaki jobban teljesít egy másik játékosnál úgy hogy a tíz hely már foglalt, akkor az alulmaradt játékos törlődik a ranglistából, és az új játékos kerül be a neki megfelelő helyre.

Osztálydiaram



Szekvencia diagram



Break Bricks

Generated by Doxygen 1.8.11

Contents

Chapter 1

Hierarchical Index

1.1 Class Hierarchy

This inheritance list is sorted roughly, but not completely, alphabetically:

Chapter 2

Class Index

2.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

Chapter 3

Class Documentation

3.1 Main.Ball Class Reference

Inheritance diagram for Main.Ball:

Public Member Functions

- Ball ()
- void move ()
- void setXDirection (int x)
- int getXDirection ()
- void setYDirection (int y)
- int getYDirection ()

Private Member Functions

- void resetState ()

Private Attributes

- int **xdirection**
- int **ydirection**

Additional Inherited Members

3.1.1 Detailed Description

A labda (Ball) elem osztálya. Amely beállítja a labda kiindulási irányát, mozgását, valamint a megfelelő képet hozzárendeli az egyedhez.

3.1.2 Constructor & Destructor Documentation

3.1.2.1 Main.Ball.Ball ( )

A Ball osztály konstruktora, megadja a labda kiindulási irányát, valamint betölti a megfelelő képet.

3.1.3 Member Function Documentation

3.1.3.1 int Main.Ball.getXDirection ( )

Az aktuális x szerinti iránnyal tér vissza.

Returns

xdirection - x tengely szerinti irány

3.1.3.2 int Main.Ball.getYDirection ( )

Az aktuális y szerinti iránnyal tér vissza.

Returns

ydirection - y tengely szerinti irány

3.1.3.3 void Main.Ball.move ( )

A labda mozgását szabályozó metódus. Ha a labda falnak ütközik, akkor visszapattan.

3.1.3.4 void Main.Ball.resetState ( ) [private]

Visszaállítja a labdát a kezdő pozicióba.

3.1.3.5 void Main.Ball.setXDirection (int *x*)

Beállítja a labda x tengely menti irányát az átvett paraméter szerint.

Parameters

x	- A paraméter ami meghatározza az xdirection értékét.

3.1.3.6 void Main.Ball.setYDirection (int *y*)

Beállítja a labda y tengely menti irányát az átvett paraméter szerint.

Parameters

y	A paraméter ami meghatározza az ydirection értékét.

The documentation for this class was generated from the following file:

- E:/Egyetem/II. ev/I. felev/Java/HF Break Bricks/src/Main/Ball.java

3.2 Main.Brick Class Reference

Inheritance diagram for Main.Brick:

Public Member Functions

- Brick (int x, int y)
- boolean isDestroyed ()
- void setDestroyed (boolean val)

Private Attributes

- boolean **destroyed**

Additional Inherited Members

3.2.1 Detailed Description

A tégla (Brick) osztály, amely beállítja az adott elemhez rendelt képet, valamint a megjelenítési pozícióját.

3.2.2 Constructor & Destructor Documentation

3.2.2.1 Main.Brick.Brick (int *x*, int *y*)

A Brick osztály konstruktora, meghatárpzza az adott tégla koordinátáit.

Parameters

x	koordináta értéke
y	koordináta értéke

3.2.3 Member Function Documentation

3.2.3.1 boolean Main.Brick.isDestroyed ( )

Returns

destroyed Visszatér a destroyed logikai változó igazságértékével (true/false).

3.2.3.2 void Main.Brick.setDestroyed (boolean *val*)

Beállitha a destroyed logikai változót a megadott igazságérték szerint.

Parameters

val	logikai változó

The documentation for this class was generated from the following file:

- E:/Egyetem/II. ev/I. felev/Java/HF Break Bricks/src/Main/Brick.java

3.3 Main.Constants Interface Reference

Inheritance diagram for Main.Constants:

Static Public Attributes

- static final int **WIDTH** = 300
- static final int **HEIGHT** = 400
- static final int **SCALE** = 2
- static final int **BOTTOM_EDGE** = 390
- static final int **NUMBER_OF_BRICKS** = 30
- static final int **START_POS_PADDLE_X** = 123
- static final int **START_POS_PADDLE_Y** = 340
- static final int **START_POS_BALL_X** = 140
- static final int **START_POS_BALL_Y** = 324
- static final int **DELAY** = 1000
- static final int **PERIOD** = 7
- static String **TITLE** = "Break Bricks"

3.3.1 Detailed Description

Created by Előd on 2016.11.25..

The documentation for this interface was generated from the following file:

- E:/Egyetem/II. ev/I. felev/Java/HF Break Bricks/src/Main/Constants.java

3.4 Main.Game Class Reference

Public Member Functions

- Game ()

Static Public Member Functions

- static void initUI ()
- static void main (String[ ] args)

3.4.1 Detailed Description

A main függvényt tartalmazó osztály, ez indítja el az alkalmazás működését.

3.4.2 Constructor & Destructor Documentation

3.4.2.1 Main.Game.Game ( )

meghívja az initUI() függvényt.

3.4.3 Member Function Documentation

3.4.3.1 static void Main.Game.initUI ( ) [static]

Ez a metódus meghatározza azt az ablakot, amelyikben a játék futni fog, és meghívja a Panel osztály konstruktorát. Tehát lényegében ennek a függvénynek a hívásával tudjuk elindítani a játékot.

3.4.3.2 static void Main.Game.main (String[ ] *args*) [static]

A main függvény. Elindítja az alkalmazás működését, egy Menu típusú változó létrehozásának a segítségével.

Parameters

args	- main függvény default paramétere

The documentation for this class was generated from the following file:

- E:/Egyetem/II. ev/I. felev/Java/HF Break Bricks/src/Main/Game.java

3.5 Main.Menu Class Reference

Inheritance diagram for Main.Menu:

Classes

- class RecordListener

Public Member Functions

- Menu ()

Static Public Attributes

- static final int **WIDTH** = 320
- static final int **HEIGHT** = WIDTH / 12 ∗9
- static final int **SCALE** = 2

Protected Attributes

- Image **image**

3.5.1 Detailed Description

Ez a Menu osztályunk, amely a User Interface megvalósítását támogatja.

3.5.2 Constructor & Destructor Documentation

3.5.2.1 Main.Menu.Menu ( )

A Menu osztályunk konstruktora, amely kirajzolja a menünket a képernyőre. Támogatja a menüpontok közötti váltázokat, valamint egy gomb megnyomása segítségével innen indítható a játékonk. Alapvetően 4 opciót tartalmaz: Play, Help, Records, Quit. A Quit gomb megnyomásával ugyan azt a funkciót érhetjük el mint a jobb felső sarokban található x kattintásával, vagyis bezáródik az alkalmazásunk. A Records menüpont alatt megtekinthető az aktuális top 10-es ranglista, amennyiben már van 10 személy a ranglistán. Ha nincs 10 személy, akkor mindenkit megjelenít, aki addig felkerült a listára. A Help menüpont alatt egy rövid útmutató/leírás található a játékhoz. A Play gomb megnyomásával pedig értelemszerűen maga a játék indítható el. A Play gombra való kattintás haátásra elindul a játék egy új ablakban.

Parameters

e	

A Help gombra való kattintás hatására megjelenik a játék rövid leírása.

Parameters

e	

A Quit gombra kattintva bezárja az alkalmazásunkat.

Parameters

e	

The documentation for this class was generated from the following file:

- E:/Egyetem/II. ev/I. felev/Java/HF Break Bricks/src/Main/Menu.java

3.6 Main.Paddle Class Reference

Inheritance diagram for Main.Paddle:

Public Member Functions

- Paddle ()
- void move ()
- void keyPressed (KeyEvent event)
- void keyReleased (KeyEvent event)

Private Member Functions

- void resetState ()

Private Attributes

- int **dx**

Additional Inherited Members

3.6.1 Detailed Description

A játékos által irányított Paddle (ütő) osztály, az irányításához és beállításához szükséges metódusokkal.

3.6.2 Constructor & Destructor Documentation

3.6.2.1 Main.Paddle.Paddle ( )

A Paddle (utő) osztály konstruktora. az egyedhez rendeli a képet, valamint beállítja a kiindulási poziciójá.

3.6.3 Member Function Documentation

3.6.3.1 void Main.Paddle.keyPressed (KeyEvent *event*)

Figyeli, hogy melyik gomb lett lenyomva. Balra nyíl esetén balra irányítja az ütőt. Jobbra nyíl lenyomása esetén pedig jobbra.

Parameters

event	KeyEvent beépített osztály paramétere

3.6.3.2 void Main.Paddle.keyReleased (KeyEvent *event*)

Figyeli a lenyomott billenytyű felengedését. Ha felengedtük a lenyomott gombot, akkor megáll az ütő egyhelyben.

Parameters

event	KeyEvent beépített osztály paramétere

3.6.3.3 void Main.Paddle.move ( )

Az utő mozgását szabályozza. Kizárólag x tengely szerint tud működni, és nem tud kimenni az ablakból. A szélénél megütközik.

3.6.3.4 void Main.Paddle.resetState ( ) [private]

Visszaállítja az ütő koordinátáit a kezdőpozícióba.

The documentation for this class was generated from the following file:

- E:/Egyetem/II. ev/I. felev/Java/HF Break Bricks/src/Main/Paddle.java

3.7 Main.Panel Class Reference

Inheritance diagram for Main.Panel:

Classes

- class ScheduleTask
- class TAdapter

Public Member Functions

- Panel ()
- void addNotify ()
- void create ()
- void paintComponent (Graphics g)
- void gameOver (Graphics2D g2d)

Public Attributes

- ArrayList< Records > **records** = new ArrayList<>()
- int **count** = 0

Private Member Functions

- void initGame ()
- void drawThings (Graphics2D g2d)
- void stopGame ()
- void checkCollision ()

Private Attributes

- Ball **ball**
- String **message** = "Game Over"
- Timer **timer**
- Timer **timer2**
- Brick **bricks** [ ]
- Paddle **paddle**
- boolean **ingame** = true

Additional Inherited Members

3.7.1 Detailed Description

Ez a Panel osztály, amely lényegében kiíratja az objektumokat a képernyőre, figyeli azok viselkedését, és ezek alapján folyamatosan újrarajzolja őket. Lényegében ez az osztály vezérli a játék működését.

3.7.2 Constructor & Destructor Documentation

3.7.2.1 Main.Panel.Panel ( )

Az osztály konstruktora, meghívja az inicialisáló metódust.

3.7.3 Member Function Documentation

3.7.3.1 void Main.Panel.addNotify ( )

Meghívja a create() függvényt.

3.7.3.2 void Main.Panel.checkCollision ( ) [private]

Vizsgálja a játék folyamata közben történt eseményeket. Ha téglámak ötküzik, akkor eltűnteti az adott téglát, a labda pedig visszapattan a megváltott iránya szerint. Ha az ütőnek ütközik, akkor is visszapattan, közben megváltoztatja irányát. Amennyiben a labda eléri az ablak alját, a játék leáll, és a játékosunk veszített.

3.7.3.3 void Main.Panel.create ( )

Létrehozza a játékhoz szükséges labdát, ütőt, valamint téglákat, azok helyének meghatározásával.

3.7.3.4 void Main.Panel.drawThings (Graphics2D *g2d*) [private]

Kirajzolja a képernyőre a megfelelő ablakba, a megfelelő helyekre a játék objektumait.

Parameters

g2d	A képernyőre történő kiírást támogatja.

3.7.3.5 void Main.Panel.gameOver (Graphics2D *g2d*)

A játék végkimenetele alapján szabályozza, hogy mi történjen azt követően. Ha a játékos veszít, akkor egy "↩ Sajnáljuk ön veszített" felirat fogadja, majd lehetősége van új játék indítására. Amennyiben a játékos sikeresen vette az akadályokat. Akkor egy "Victory" felirat után megadhatja a nevét, amelyet a program egy ranglistába tárol a játékidő alapján.

Parameters

g2d	A képernyőre történő kiírást támogatja.

3.7.3.6 void Main.Panel.initGame ( ) [private]

Ez a metódus inicializálja a játékhoz szükséges változókat. Megadja a téglák számát, elindít egy belső timert, vagyis esetünkben kettőt. Egyiket a játék periódusidejével, egy másikat pedig 1000ms -os periódusidővel, hogy meg tudjuk határozni a játékos játékban töltött idejét.

3.7.3.7 void Main.Panel.paintComponent (Graphics *g*)

Ha játékban vagyunk, akkor hívja a drawThings() metódust, ha a játéknak vége akkor pedig a gameOver() metódust.

Parameters

g	A képernyőre történő kiírást támogatja.

3.7.3.8 void Main.Panel.stopGame ( ) [private]

Leállítja a játékot.

The documentation for this class was generated from the following file:

- E:/Egyetem/II. ev/I. felev/Java/HF Break Bricks/src/Main/Panel.java

3.8 Main.Menu.RecordListener Class Reference

Inheritance diagram for Main.Menu.RecordListener:

Public Member Functions

- void actionPerformed (ActionEvent e)

3.8.1 Detailed Description

Segédosztály, a Record gomb megnyomásával kiváltott történések támogatására.

3.8.2 Member Function Documentation

3.8.2.1 void Main.Menu.RecordListener.actionPerformed (ActionEvent *e*)

Ha a menüben rákattintottunk a Records gombra, akkor betölti az annak megfelelő ablakot, így láthatjuk és kiovlashatjuk a toplistát.

Parameters

e	

The documentation for this class was generated from the following file:

- E:/Egyetem/II. ev/I. felev/Java/HF Break Bricks/src/Main/Menu.java

3.9 Main.Records Class Reference

Inheritance diagram for Main.Records:

Public Member Functions

- Records (String name, int score)
- void writeObject (ObjectOutputStream oos)
- void readObject (ObjectInputStream ois)
- String getName ()
- int getScore ()
- int compareTo (Records r)

Private Attributes

- String **name**
- int **score**

3.9.1 Detailed Description

Lényegében egy segédosztály, amely a ranglistánk fájlba írását, illetve fájlból történő beolvasását támogatja.

3.9.2 Constructor & Destructor Documentation

3.9.2.1 Main.Records.Records (String *name,* int *score*)

Az osztályunk konstruktora, meghatározza a paramétereit, a megadott értékek alapján.

Parameters

name	A játékos nevét tárolja.
score	A játékoshoz és az adott játékhoz tartozó időt méri

3.9.3 Member Function Documentation

3.9.3.1 int Main.Records.compareTo (Records *r*)

Az adatok count paramétere szerinti sorrendezett tárolást támogató metódus.

Parameters

r	Egy Record típusú változó

Returns

-1, ha az aktuális idő nagyobb mint amihez épp hasonlítjuk 1, ha kisebb 0, különben.

3.9.3.2 String Main.Records.getName ( )

Getter: a name paraméter értékével tér vissza.

Returns

name - Visszatér a játékos nevével.

3.9.3.3 int Main.Records.getScore ( )

Getter: a score paraméter értékével tér vissza.

Returns

score - Visszatér a játékos idejével.

3.9.3.4 void Main.Records.readObject (ObjectInputStream *ois*)

Egy adott rekord (mint egyed/elem) fájlból történő beolvasását valósítja meg.

Parameters

ois	ObjectInputStream paraméter, a fájlból olvasást támogatja.

3.9.3.5 void Main.Records.writeObject (ObjectOutputStream *oos*)

A kapott elemhez tartozó adatok fájlba írását valósítja meg ez a metódus.

Parameters

oos	ObjectOutputStream paraméter, a fájlba írást támogatja.

The documentation for this class was generated from the following file:

- E:/Egyetem/II. ev/I. felev/Java/HF Break Bricks/src/Main/Records.java

3.10 Main.Panel.ScheduleTask Class Reference

Inheritance diagram for Main.Panel.ScheduleTask:

Public Member Functions

- void **run** ()

3.10.1 Detailed Description

Minden PERIOD ms-ban meghívódik. A run() metódusával mozgatjuk az ütőt és ezáltal a labdát is. Mindeküzben figyeljük az esetleges ütözéseket, és újrarajzoljuk az ablak tartalmát.

The documentation for this class was generated from the following file:

- E:/Egyetem/II. ev/I. felev/Java/HF Break Bricks/src/Main/Panel.java

3.11 Main.Panel.TAdapter Class Reference

Inheritance diagram for Main.Panel.TAdapter:

Public Member Functions

- void keyReleased (KeyEvent event)
- void keyPressed (KeyEvent event)

3.11.1 Detailed Description

Segédosztály, amely figyeli hogy mikor nyomjuk le és engedjük fel a paddle-t irányító gombot.

3.11.2 Member Function Documentation

3.11.2.1 void Main.Panel.TAdapter.keyPressed (KeyEvent *event*)

Meghívja a Paddle osztály keyPressed osztályát.

Parameters

event	KeyEvent beépített osztály paramétere

3.11.2.2 void Main.Panel.TAdapter.keyReleased (KeyEvent *event*)

Meghívja a Paddle osztály keyReleased osztályát.

Parameters

event	KeyEvent beépített osztály paramétere

The documentation for this class was generated from the following file:

- E:/Egyetem/II. ev/I. felev/Java/HF Break Bricks/src/Main/Panel.java

3.12 test.Test2 Class Reference

Public Member Functions

- void TestWizardSetAndGetX ()
- void TestWizardSetAndGetY ()
- void TestBallSetAndGetXDirection ()
- void TestBallSetAndGetYDirection ()
- void RecordReadTest () throws IOException
- void TestBrickIsDestroyedAndsetDestroyed ()

3.12.1 Detailed Description

Created by Előd on 2016.11.29..

3.12.2 Member Function Documentation

3.12.2.1 void test.Test2.RecordReadTest ( ) throws IOException

A Record osztály readObject() függcényének a tesztelése, amely egy fájlból olvas be adatokat, amennyiben a fájl létezik. Viszotn ha nem létezik a megadott nevű fájl, IOExeptiont dob.

Exceptions

IOException	- IO kivételt dob hiba esetén.

3.12.2.2 void test.Test2.TestBallSetAndGetXDirection ( )

A Ball osztály setXDirection(), illetve getXDirection() függvényeinek tesztelése.

3.12.2.3 void test.Test2.TestBallSetAndGetYDirection ( )

A Ball osztály setYDirection(), illetve getYDirection() függvényeinek tesztelése.

3.12.2.4 void test.Test2.TestBrickIsDestroyedAndsetDestroyed ( )

A Brick osztály isDestroyed(), illetve setDestroyed() függvényeinek tesztelése.

3.12.2.5 void test.Test2.TestWizardSetAndGetX ( )

A Wizard osztály setX(), illetve getX() függvényeinek tesztelése.

3.12.2.6 void test.Test2.TestWizardSetAndGetY ( )

A Wizard osztály setY(), illetve getY() függvényeinek tesztelése.

The documentation for this class was generated from the following file:

- E:/Egyetem/II. ev/I. felev/Java/HF Break Bricks/test/test/Test2.java

3.13 Main.Wizard Class Reference

Inheritance diagram for Main.Wizard:

Public Member Functions

- void setX (int x)
- void setY (int y)
- int getX ()
- int getY ()
- int getWidth ()
- int getHeight ()

Public Attributes

- Image **image**

Protected Attributes

- int **x**
- int **y**
- int **iwidth**
- int **iheight**

3.13.1 Detailed Description

A Wizard(magyarul: varázsló, későbbiekben is fogok rá így hivatkozni) osztály egy segédosztály, amit felhasználnak az elemek osztályai. Itt vannak azok az változók és metódusok, amiket a Brick, Ball, Paddle osztályok, valamint objektumok egyaránt felhasználnak. Úgymond egy segédosztály ami megkönnyti a többi osztály működését, és nem kell ezeketa metódusokat minden osztályba külön definiálni, megkönnyítve a program átláthatóságát és fejleszthetőségét.

3.13.2 Member Function Documentation

3.13.2.1 int Main.Wizard.getHeight ( )

A hozzá tartozó kép magasságát határozza meg.

Returns

iheight Visszatér az iheight értékkel

3.13.2.2 int Main.Wizard.getWidth ( )

Visszatér az iwidth értékkel, ami a hozzá tartozó kép szélességét határozza meg.

Returns

iwidth

3.13.2.3 int Main.Wizard.getX ( )

Visszatér az aktuális x paraméterrel.

Returns

x koordináta értéke

3.13.2.4 int Main.Wizard.getY ( )

Visszatér az aktuális y paraméterrel.

Returns

y koordináta értéke

3.13.2.5 void Main.Wizard.setX (int *x*)

Beállítja az x paramétert a megadott értékre.

Parameters

x	az x koordináta értéke

3.13.2.6 void Main.Wizard.setY (int *y*)

Beállítja az y paramétert a megadott értékre.

Parameters

y	az y koordináta értéke

The documentation for this class was generated from the following file:

- E:/Egyetem/II. ev/I. felev/Java/HF Break Bricks/src/Main/Wizard.java

Index