Име : Петко Пенчовски Предмет: Програмиране с Java,част 2

Име : Георги Пенчовски

Дата : 14.04.2018

email : busec@abv.bg, [fostins@abv.bg](mailto:fostins@abv.bg)

GitHub : <https://github.com/georgipenchovski/SpaceInvadersJava>

*SPACE INVADERS*

**1. Условие**

Условието на проекта е изготвянето на аркадната игра от 1978 – ***Space Invaders*** .Оригиналът е създаден от ***Tomohiro Nishikado****.* В нашия случай играта ще бъде направена в оригинален ‘’***skin***’’ ***Star Wars*** .

**2. Въведение**

Програмата е реализирана на платформата ***Java***.

**3. Теория**

***Space Invaders*** е аркадна видео игра, създадена от ***Tomohiro Nishikado*** и издадена през 1978 година. Първоначално е произведена и продадена от ***Taito*** в Япония и по-късно е лицензирана за производство в Съединените щати от ***Midway Division*** на ***Bally***. ***Space Invaders*** е една от най-ранните игри за стрелба и целта е да се победят вълни от извънземни с лазерно оръдие, за да спечелят колкото се може повече точки. Колко по-близо идват нашествениците, толкова по- бързо започват да се движат. При проектирането на играта ***Nishikado*** се е вдъхновил от популярните медии: Пробив, Войната на световете и Междузвездни войни.

**4. Инсталация и настройки**

Необходима е инсталирана **Java** на компютъра, след което се отива в папка **src** ,в която има папка ***main*** и се стартира файла ***main*** .

**5. Описание на програмния код**

Кодът се състой от 4 главни пакета:

* ***main***
* ***images***
* ***sounds***
* ***Space\_Invaders***

В ***main*** методът, класа ***SwingUtilities.invokeLater(new Runnable()*** създава пример за анонимно внедряване на интерфейса ***Runnable*** и го предава на ***invokeLater.*** В методът ***public void run()*** се създава рамката на играта и се подкарва с вградения от ***Java*** метод ***.start.***

Пакетът ***images*** съдържа изображенията, а в пакета ***sounds*** са звуците.

В пакета ***Space\_Invaders*** са пакетите ***objects*** и ***core,*** и два класа***.***

Пакет ***core:***

* ***class BitmapFontRenderer -*** това е класа за изобразяване на шрифтове. Този клас взима картинката(***images/font10x10.png***) , която е зададена в абстрактния ***class*** ***Game*** и се запазва в матрица 16х16,като матрицата е декларирана като поле в ***class BitmapFontRenderer*** ,а размерите и са в ***class***  ***Game .***
* ***class Display -*** наследява ***class JPanel.*** В този клас се задават размерите на рамката и цвета на фона.
* ***abstract class Game -*** в него са направени методи, с определени функции, които се използват в другите класове.
* ***class Keyboard -***методът ***public void keyPressed()*** активира функцията на бутона, когато е натиснат, а ***public void keyReleased()*** деактивира функцията на бутона, когато не е натиснат.
* ***abstract class Obj -*** В този клас се инициализират проментиливи, които зареждат картинки, правят корабите и изстрелите в играта видими,и да унищожават.Тези променливи се използват в почти всички класове на проекта. методът ***public void draw (Graphics2D g)*** отговаря за смяната на кадъра. ***public void updateCollider ()-***прави движението на лазерите, а ***public void***  ***loadFrames()*** зарежда картинките на корабите в състояние убит и жив, и изтрелите им.
* ***class Sound -*** е създадена променлива ***soundclip*** от типа ***interface Clip*** и се използват готовите интерфейси - ***start***, ***loop*** и ***stop*** за манипулация на звуците.

Пакет ***objects:***

* ***class Alien -*** съставен е от методите, ***public void init()*** - задава зареждането и изчезването на извънземните, и звукът при тяхното изчезване.***public void updateReady()*** - показва извънземните в началото на играта.***public void updatePlaying()*** - скоростта на движение на извънземните при намалянето на техния брой.***public void stateChanged()*** - определя състоянието на играта, спрямо извънземните, ако има живи, играта продължава и ако са убити или са стигнали до долу, играта свършва. ***public void reset()*** -подрежда извънземните. ***public void hit()*** - ако е уцелено извънземно го премахва със съответния звук и дава точки. ***public void downEnemy().public void frameEnd()*** - ограничава извънземните да не излязат от рамката.
* ***class AlienShot - public void init() -*** определя обхвата на унищожение на изтрела и зарежда картинките и прави изстрела да убива кораба ,а не да минава през него. Това става става с променливата ***collider*** от класа ***Rectangle2D***. ***public void updatePlaying()*** - времето в, което изстрелите се задържат на екрана и тяхната скорост(техния интервал). ***public void canShoot()*** - прави извънземното да стреля. ***public void shoot()*** - изстрелът и неговите координати. ***public void stateChanged()*** - не се ресетват изстрелите. ***public void hit()*** - интервал на изстрела.
* ***class GameOver - public void updateGameOver() -*** изчиства натрупаните точки и ни записва в най-добрия резултат и връща състоянието на играта в начална позиция преди стартиране. ***public void draw()-*** изписва финалния надпис ''**GAME OVER**''. ***public void stateChanged()*** - прави, така че всичко направено в съответния клас да бъде видимо в играта.
* ***class Inscriptions - public void hit()*** - зарежда картинката за животите на кораба. ***public void draw()-*** определя позициите на ''SCRORE'', ''HI-SCORE'' и ''LIVES'' и задава цвета, дебелината, и позицията на линията под кораба. ***public void stateChanged ()*** - прави, така че всичко направено в съответния клас да бъде видимо в играта.
* ***class Initializer- public void init()-*** стартира се играта с него***.public void updateInitalizing()*** -прави ,така че всички обекти от кода, без музиката да тръгнат в играта и е ''Override'' с метод ***public void updateInitalizing()*** вкласа ***SpaceInvadersObj. public void stateChanged()*** .
* ***class PenchovskiPresents - public void init()-*** определя размера на началния надпис ***''PENCHOVSKI BROS''.public void updateОlPresents()-*** пуска началния надпис ***''PENCHOVSKI BROS''.public void draw()-*** размер на текста***. public void stateChanged ()*** - прави, така че всичко направено в съответния клас да бъде видимо в играта.
* ***class DeathStar* -** това е класа за "Звездата на Смъртта", която се появява най-отгоре. ***public void init()-*** зарежда картинката на "Звездата на Смъртта", на унищожена и на точките.***public void updatePlaying()*** - интервала на появяване на "Звездата на Смъртта" и ако е ударен , се появяват точките. ***public void stateChanged ()*** - прави, така че да се появява отново, след като бъде уцелена веднъж. ***public void reset()-*** метод, който се използва за ресетване на промяната на "Звездата на Смъртта". Използва се да смени картинките на "Звездата на Смъртта", когато е уцелена. ***public void hit()*** - прави, така че да може да бъде ударена от изстрел от нашия кораб.
* ***class Shield - public void init()-*** определя позицията на щитовете върху екрана и зарежда картинките за тях. ***private void allNeigborsShieldHit() -*** прави, така че щитовете да могат да изчезват напълно след като бъдат ударени от изстрел. ***public void updatePlaying()*** - ако щита е ударен, да се чупи.***public void stateChanged ()*** - прави, така че всичко направено в съответния клас да бъде видимо в играта. ***public void hit()*** - щита се чупи равномерно при удар.
* ***class Ship - public void init()-***  определя позицията на кораба и зарежда картинките в различни състояния( унищожен и начално положение). ***public void updatePlaying()*** - приема командите от клавиатурата за наляво, надясно и изстрел. ***public void updateClear()*** - при превъртане на нивото методът позволява да се изграе още една игра. ***public void updateHit()*** - метода премахва кораба, ако е ударен. ***public void stateChanged ()*** - прави, така че всичко направено в съответния клас да бъде видимо в играта.
* ***class Shipshot - public void init()-***  прави, така че изстрела да убива пришълците с ***collider,*** зарежда картинките и звук. ***public void updatePlaying()*** - ресетва изстрелите и ги прави да унищожават. ***public void canShoot()*** - прави кораба да стреля. ***public void shoot()*** - картинката и звука на изстрела. ***public void stateChanged ()*** - когато е приключила играта изстрела не се вижда. ***public void hit()*** - интервал на изстрела.
* ***class Title - public void init()-***  зарежда надписа***''STAR WARS''***  и определя големината му. От този метод се извиква и началната мелодия. ***public void updateTitle()-***  при натискане на ***SPACE*** мелодията спира и играта започва. ***public void draw()-*** размер на текста. ***public void stateChanged ()*** - прави, така че всичко направено в съответния клас да бъде видимо в играта.
* ***class SpaceInvadersGame -*** В класа са зададени размери на рамката и мащаб на играта в нея. ***public State getState()*** и ***public void setState()-*** гетер и сетер***. public void addScore()-*** добавя точки. ***public void clearScore()-*** ресетва резултата. ***public void updateHiScore() -*** ъпдейтва ***HiScore. public void allObjects() -*** с три вложени цикли са одредени циклите на екрана, зададени са броят на типовете извънземни в масив и броят на точките, които дават. ***public void getScreenSize()-*** гетер***. public void getScreenScale() -*** гетер. ***public String getScore()-*** прави, така че да се променя надписа на точките като се увеличават. Превръща ***String*** в масив и му добавя точки. ***public String gHiScore()-*** прави, така че да с променя надписа на ***HiScore*** . Превръща ***String*** в масив и му добавя точки. ***public void startGame()-*** метод за започване на играта. ***public void gameCleared()-*** свършва играта. ***public void nextGame()-*** ресетва играта.
* ***class SpaceInvadersObj -*** В този клас методите са ***Overridden*** в другите класове.

**6. Приноси на курсистите, ограничения и възможности за бъдещи разширения**

Играта може да бъде направена с по-добра графика и звук.

**7. Източници**

[https://www.youtube.com/watch?v=0RSqbDKbnXc HYPERLINK "https://www.youtube.com/watch?v=0RSqbDKbnXc&list=PL8MfEpjBKpQTG7WZMWM0FA9dWjHc40dxU"& HYPERLINK "https://www.youtube.com/watch?v=0RSqbDKbnXc&list=PL8MfEpjBKpQTG7WZMWM0FA9dWjHc40dxU"list=PL8MfEpjBKpQTG7WZMWM0FA9dWjHc40dxU](https://www.youtube.com/watch?v=0RSqbDKbnXc&list=PL8MfEpjBKpQTG7WZMWM0FA9dWjHc40dxU)

[https://www.youtube.com/watch?v=cORWVUL6eg4 HYPERLINK "https://www.youtube.com/watch?v=cORWVUL6eg4&t=18s"& HYPERLINK "https://www.youtube.com/watch?v=cORWVUL6eg4&t=18s"t=18s](https://www.youtube.com/watch?v=cORWVUL6eg4&t=18s)

<http://howtocarveroastunicorn.blogspot.bg/2013/05/old-star-wars-old-atari-good-old-retro.html>