

READ ME:

Name: Lee Alima

ID: 313467441

INTERFACES AND CLASSES IN THE CODE:

Enemy

I created an "Enemy" class that extends Block. In my opinion, an enemy is a kind of a Block and has its properties. Moreover, the "Enemy" object should have the ability to move.

methods : the methods of Block and , copyEnemy – a method that created a new copy of the enemy and returns it, moveBy – a method that moves the enemy by movement, drawOn – new method to draw an image, getRectangle – a method that returns the enemy's rectangle, returnRect – as a method that returns the rectangle from the constructor and setRectangle – a method that set the rectangle coordinates.

EnemyMovement

this class is the kind of sprite, this method is in charge of the "matrix" movement.

this class has the ability to arrange the list of chickens in columns.

methods : methods – the sprite methods, and findLastInList– this method uses the member of list of chickens to a list of the last chicken in every column, createLinkedLists – this method creates a sort of matrix to represent the chickens and arrange them into columns, updateLinked – this method handle with a deletion in the list, shoot – this method is responsible of the chicken's shooting, startAgain – this method set the chickens at the start location, findMaxX – to find the maximum x coordinated after the the movement, findMinX – to find the minimum x coordinate after the movement, touchTheShield - to check if the chickens touches the shield and addToGame - to add the current object to the game.

SpaceInvadersLevel

a class the implements LevelInformation.

this method creates and set the elements for a level.

this object is being created in gameFlow for each level – but with different speed.

methods: this class has the methods of LevelInformation and getImage – to read the image of the background.

BackgroundB

this class is in type of Sprite.

methods: the methods of sprite

** I change the gameFlow class so now, for every level it creates a GameLevel object with a new LevelInformation object I sent to him. With every win of level – I run a level which is harder than the previous one.

IMPLEMENTATION:

The Aliens formation:

In the class of EnemyMovement I use the list of chickens and I arrange the chickens in a list of linked list – every linked list represents a column in the matrix. In that way I have 10 columns of 5 chickens in each. In order to move the chickens I check if the rightmost and leftmost chickens can move – if they can move I moved them, else I lowered the chickens. in case of deletion I update the matrix and the list of chickens.

The Shields:

I created a loop inside a loop in order to create a matrix of blocks.

Shots by chickens:

I handle the creation of shooting inside the doOneFrame in GameLevel – I send the game to enemyMovement and there I created the shoots. I check the time of the shooting and the current time – if the difference between them is 0.35 seconds I called the shot method again.

Shots by player:

I send the game to Paddle – there in timePassed I check the time as I check for the shooting from the chickens. I called the shoot method and created a ball and shoot it from the paddle.