Development of a game software using the "Processing" language

Sergiu BEŞLIU, Mihail KULEV
Technical University of Moldova
Sergiu.V.Besliu@gmail.com , mkmk54@mail.ru

Abstract — **The game software was developed as a course work assignment in my "Data Structures and Algorithms" class using the "Processing" language and based upon the principles of Object-Oriented Programming .**

Index Terms — **Game , Object-Oriented Programming , Processing**

I. INTRODUCTION

I developed a game software called "Invading Spacers" in the "Processing" language . It is based on the hit arcade game "Space Invaders" .
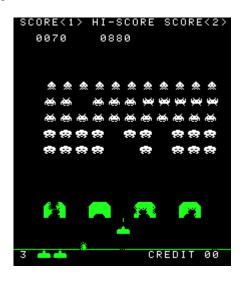

Fig. 1 Screenshot of the "Space Invaders" game

It is a two-dimensional shooter game . The player controls a ship which can move and fire projectiles at a fixed interval . The objective is to destroy 3 rows of aliens which are slowly advancing towards the player's ship and also shooting at it . If the player's ship is hit , the game ends . For this purpose , I defined classes for 4 entities in my program : Ship , Alien , Bullet and Timer , which ,combined , have 14 functions . I also used certain images to visualise the Ship and Alien objects on screen .

II. PROGRAM DESCRIPTION

The "Ship" class defines the player's ship object . The ship is displayed in the lower part of the screen and can only move horizontally . It is controlled by moving the mouse and shooting is done by clicking the left or right mouse button .

The "Alien" class represents the alien ships' objects . The alien ships are displayed in the upper part of the screen but advance towards the bottom of the screen while shooting projectiles randomly towards the player . They move at a fixed speed and shoot at an interval which gradually decreases , thereby challenging the player to destroy them faster .

The "Bullet" class represents the projectiles shot by both the alien ships and the player's ship . They are displayed as small colored rectangles which move vertically upwards or downwards at a fixed speed , depending on which ship they were shot by . They are colored green and move downwards when shot by aliens , and red and move upwards respectively when shot by the player's ship .

The "Timer" class is an auxiliary class used for various purposes , such as setting the interval between the shots fired by the player's and aliens' ships .

III. PROGRAM VISUALISATION

I will show some screenshots of my game software which describe certain situations within the game .
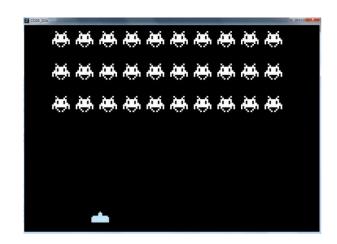

Fig . 2 Screenshot showing the aliens' ships (top) and

player's ship (bottom) at the beginning.

As you can see there are three rows of aliens placed at a certain distance from each other .

Fig. 3 Screenshot showing the game in action , with the player having destroyed a few aliens .

Here the player has already destroyed a few alien vessels. The movement pattern of the aliens is hard to observe here . They move horizontally back and forth , but once they reach a margin of the screen , they shift their positions down a bit , so they slowly progress towards the bottom .

Fig. 4 Screenshot showing the end of the game message once the player's ship has been hit .

Finally , the player's ship was (unfortunately) hit by an alien projectile , so the game ends . The frame freezes and a "GAME OVER" text is displayed on-screen .

If the player manages to destroy all the alien ships , then everything is reinitialized and the game starts again , with one exception . The interval between the aliens' shots is not reinitialized . This makes the game harder and provides a kind of "level" system .

IV. CONCLUSION

By developing this (relatively) simple game software , I gained some experience in programming as a whole . I understood the basics of object-oriented programming, which I used and also learned to develop basic game subroutines such as one for detecting the collision between the projectile and either the player's or the aliens' vessels . I learned to write code in the Processing language , which is an excellent one for beginner programmers , or those looking for a way to easily implement graphics in a program .

ACKNOWLEDGMENTS

REFERENCES

[1] Daniel Shiffman ,Learning Processing : A Beginner's Guide to Programming Images , Animation and Interaction , Morgan Kaufman Series in Computer Graphics , 2008 .