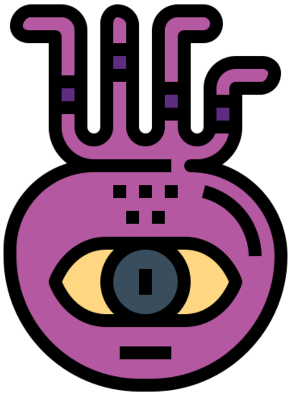
UNIVERSITATEA TEHNICĂ
DIN CLUJ-NAPOCA

Space Invaders



Student: Jurcan Diana-Maria

Group: 30422

Year: 2

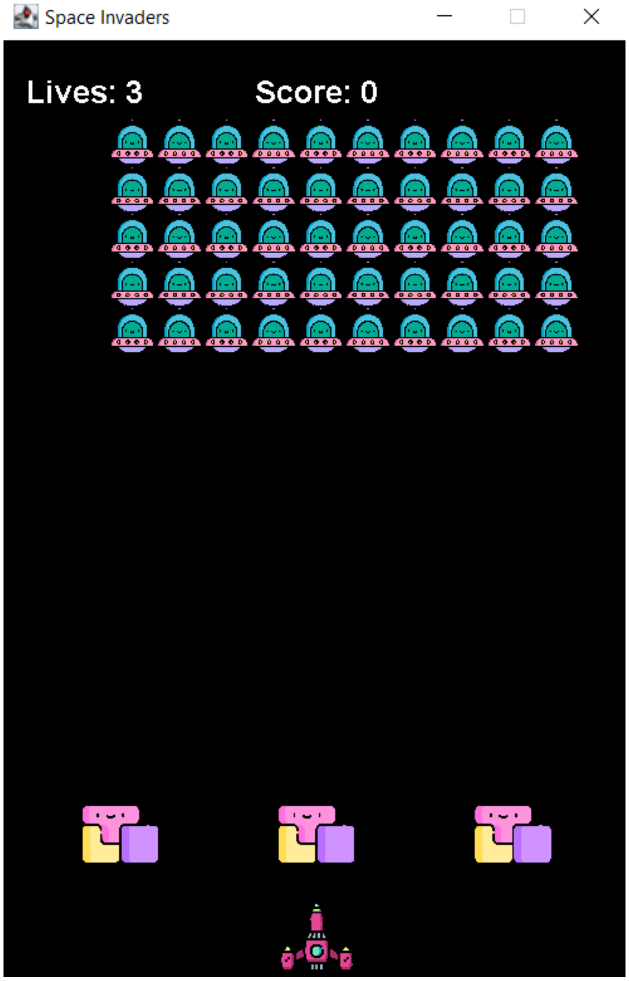
Coordinating professor: Ioan Solovăstru

Table of contents

1. Short description

Space Invaders is a fixed shooter in which the player moves rocket horizontally across the bottom of the screen and fires at aliens overhead. The aliens begin as five rows of ten that move left and right as a group, shifting downward each time they reach a screen edge.

The goal is to eliminate all of the aliens by shooting them. While the player has three lives, the game ends immediately if the invaders reach the bottom of the screen. The aliens attempt to destroy the player by firing projectiles. The rocket is partially protected by stationary defensive obstacles.

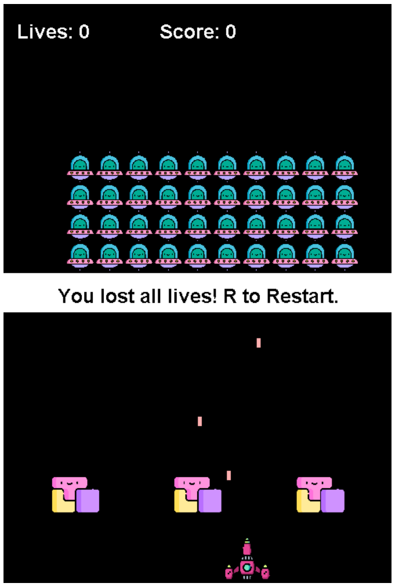


2. List of features

- The player controls a rocket that can move left/right, which will be placed at all times at the bottom of the screen. Fire projectiles can be sent upward by pressing space.
- Aliens begin as five rows of ten that move uniformly left and right as a group, shifting downward each time they reach a screen edge.
- The aliens, also, fire projectiles at random intervals of time.
- Both the projectile and alien are destroyed, when projectiles hit aliens.
- The projectile is destroyed, when projectiles hit obstacles.
- The player loses a life, when projectiles hit the player.

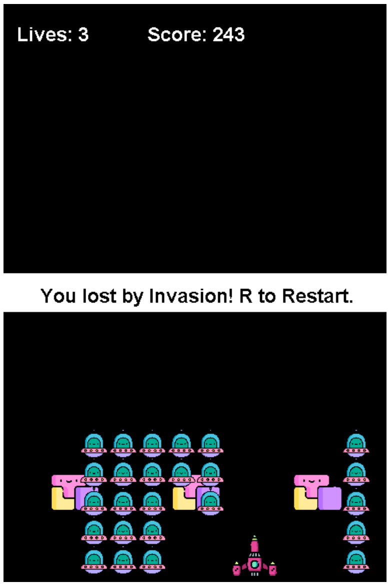
- The game ends under three conditions.
  1. The user loses if the player runs out of lives.
  2. The user loses if the aliens reach the bottom of the screen and touch the player.
  3. The user wins if all aliens are destroyed.

All of those cases can be seen below, as follows:



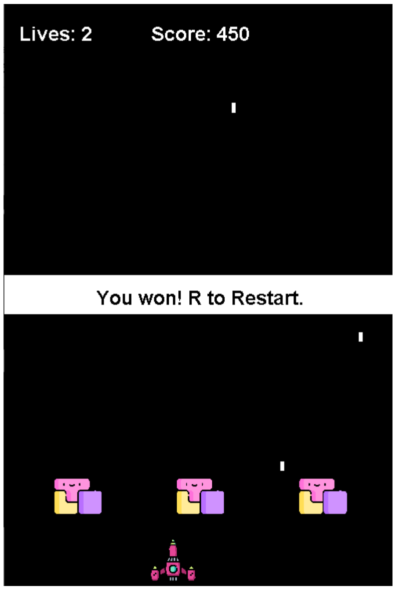
Lives: 0 Score: 0

You lost all lives! R to Restart.



Lives: 3 Score: 243

You lost by Invasion! R to Restart.



Lives: 2 Score: 450

You won! R to Restart.

3. Further Implementation

In this section, the classes used for the development of the game will be presented briefly:

- **Game:** Creates the JFrame and receives the keyboard input.
- **GamePanel:** Shows the current game stats, controls the ObjectManager, game state, and keeps the timer going for regular updates on set intervals.
- **ActionTimer:** Tracks time between firing projectiles.
- **Position:** A position (X and Y coordinates).
- **Rectangle:** Defines an area through width, height and position.
- **Alien:** Defines the alien which can fire projectiles on random intervals of time.
- **AlienManager:** Manages the moving of all the aliens.
- **CollidableObject:** An interface used to make objects have a hit() action to be called when being hit by a projectile.
- **ObjectManager:** Controls the player, obstacles, aliens, and projectiles.
- **Obstacle:** Blocks projectiles.
- **Player:** The rocket that can be moved left/right and fire shots with space.
- **Projectile:** The object that moves up or down from the Alien or Player.