

CS4303 P1 - Report

190022658

February 17, 2023

1 Overview

The practical tasked us to implement in Processing a variant of the classic Artillery Game. The game is made attractive using colors and clouds. The physics components of the implementation contains

- Damping force
- Gravity
- Wind

The controls include:

- moving the tank using 'a' and 'd'
- control angle using right and left arrow keys
- control strength using up and down arrow keys
- restarting the game at the end using the enter key

2 Design

This is a program written in the Processing language for a two-player tank game. The game involves two tanks shooting at each other until one tank's health points are reduced to zero. The tanks take turns firing at each other, and the player who manages to eliminate the opponent's tank first wins the game.

The game screen is divided into two parts, with the terrain at the bottom of the screen, and the tank and obstacle on top of the terrain. The tanks can move horizontally using the 'a' and 'd' keys, and can adjust the angle

and power of their shots using the arrow keys. Pressing the spacebar fires the projectile, and the wind speed and direction affect the trajectory of the projectile. The game ends when one tank is destroyed, and the player can restart the game by pressing the Enter key.

The program defines the game objects, such as the tanks and obstacles, as classes that inherit from the `GameObject` class. The game objects are stored in `ArrayLists`, which allow for easy iteration and modification of the objects. The game logic is implemented using a series of conditional statements and boolean flags that track the state of the game. The physics simulation of the projectile is implemented using the `PVector` class, which represents a two-dimensional vector, and a damping factor that simulates air resistance.

The game state is stored as an enum. This checks at key presses and when drawing on screen. This creates an opening screen, a playing screen, and a closing scene where you can restart the game.

Overall, this program provides an example of how to implement a simple 2D game using the Processing language.

3 Testing

4 Conclusion

This was a very good opportunity to learn the Processing language, and implement some of the concepts from the Physics component of the module. The practical was also very fun to implement. If I had more time I would have written a more sophisticated AI. I would also design the game in a better way.



Figure 1: Starting screen

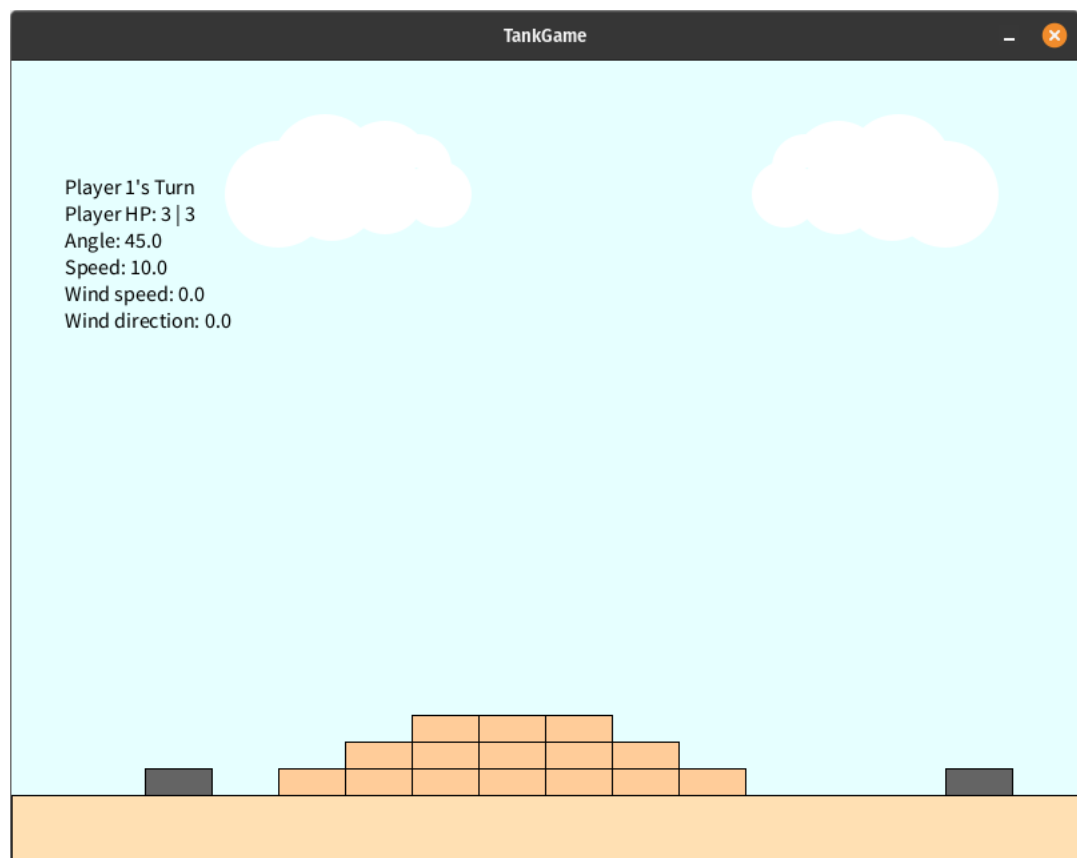


Figure 2: Starting playing

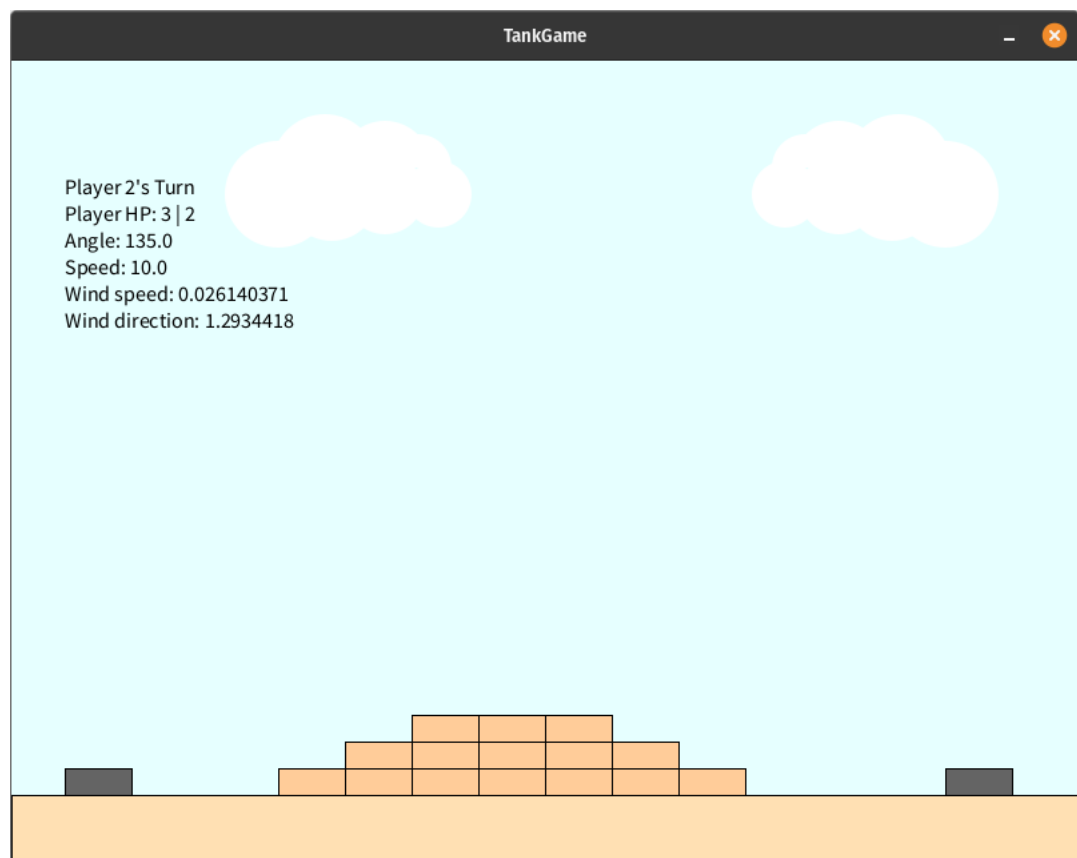


Figure 3: HP decreasing

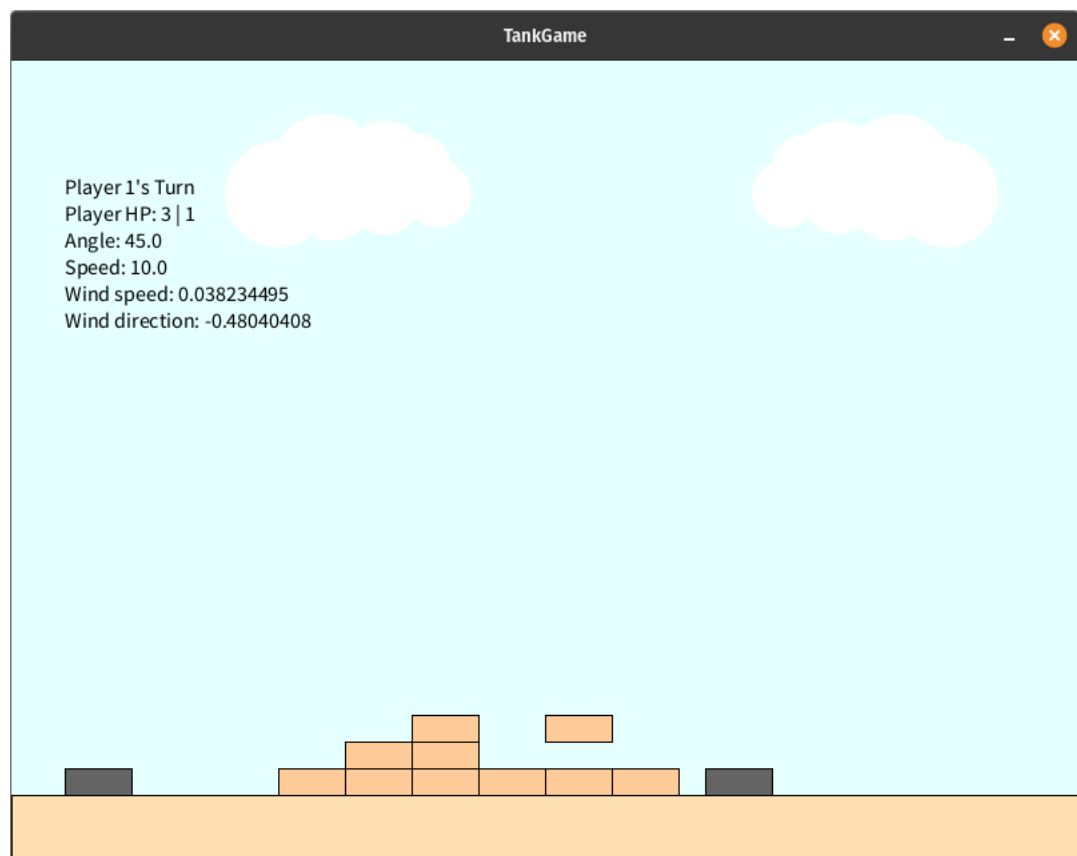


Figure 4: After a while

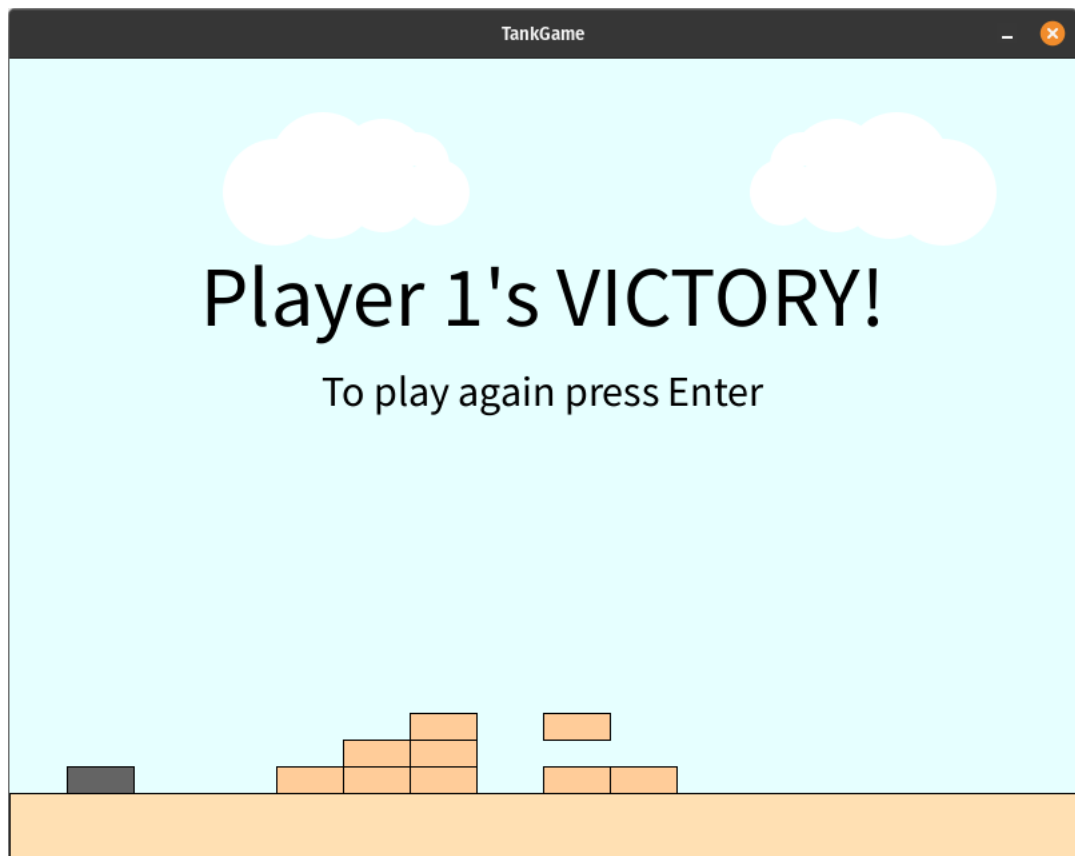


Figure 5: Game over