Wesley Gibbs
Elijah Priestley
Joshua Sherwood

Final Project Report

1. What is the goal of the game?

The goal of this game is for the player to continue to eat apples with their snake character and grow larger until they finally fill up the entire screen.

2. What are the steps to achieve the goal?

The player must move their snake character around the screen without running into the border that runs along the square edges of the playable area, or running into themselves. The player must continue to grow larger throughout the game until they are finally able to fill the screen with their snake character. When they finally do so, they have won the game, as there are no other spaces left to fill with the snake.

3. How do you communicate these steps with the player of the game?

The simple design of the game causes the steps to be easily determined. As soon as an arrow key is pressed, the player realizes that the means of winning must be navigating around the screen. As soon as an apple is eaten, the snake grows, and a new apple appears; it becomes clear that the objective is to consume as many apples as possible.

4. Is the game about luck, skill or both?

I would say that this game is almost entirely about skill. This game is a copy of the popular game “Snake,” in which the player's goal is to move their snake character around and eat apples to grow longer. As the game progresses, the player becomes larger and larger. This works against the player as it inhibits their ability to move freely. In the end, there are very few randomly generated events that could overwhelm a player's raw skill as it pertains to avoiding a loss condition. The player can only lose if they run into the border of the playable area or if they run their playable snake character into themselves. If played perfectly every single time, it would be seemingly impossible to lose based solely on luck or chance.

5. What game patterns does the game use?

This game doesn’t seem to perfectly match any patterns from the provided video; however, the main reason it is fun is due to a variation of the “social status” pattern. Rather than competing against other players, though, this game is a competition against oneself. The more attempts a player makes to fill the screen, the better they will get at the game, motivating them to continue competing against themselves in order to get a higher score than in previous attempts.

DocStrings

MODULE DOCSTRING:

Simple Wormy game made with PyGame.

This program runs a basic snake-style game where a worm moves around, eats apples, and grows longer. The game ends if the worm hits the wall or runs into itself.

main:

Sets up the game window and keeps the main game loop running. Shows the start screen, plays a round, and then shows the game-over screen.

runGame:

Run one full round of the game.

Handles movement, collisions, score updates, and drawing everything on the screen until the player loses the game.

drawPressKeyMsg:

Draw a small message telling the player to press a key to begin.

checkForKeyPress:

Check if the player released a key.

Returns the key that was released, or none if nothing was pressed. Also quits the game if needed.

showStartScreen:

Show the starting title screen with rotating text. Waits until the player presses a key to continue.

terminate:

Close pygame and exit the program.

getRandomLocation:

Return a random x and y spot on the grid to place an apple.

showGameOverScreen:

Show the 'Game Over' screen and wait for the player to either play again or quit the game.

drawScore:

Draw the player's score in the top-right corner.

score: the number of points to show.

drawWorm:

Draw the worm on the screen using the list of its coordinates.

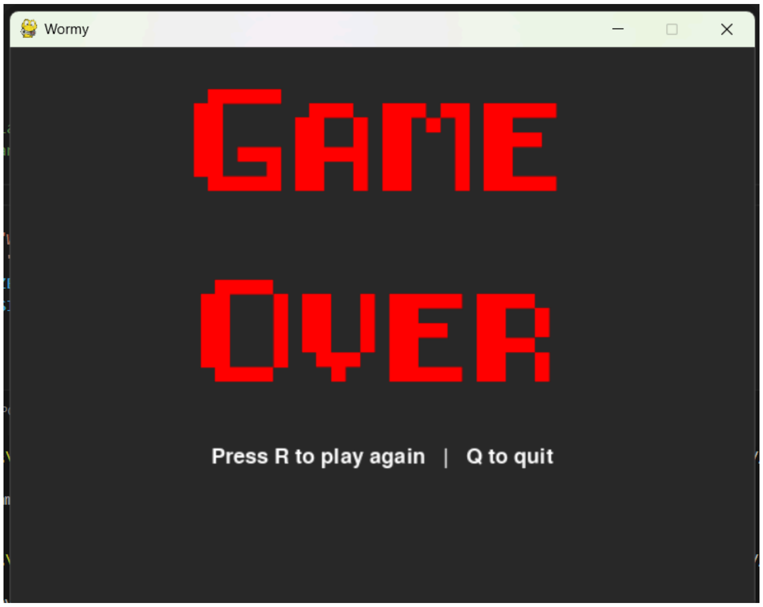
drawApple:

Draw the apple at the given grid location.

drawGrid:

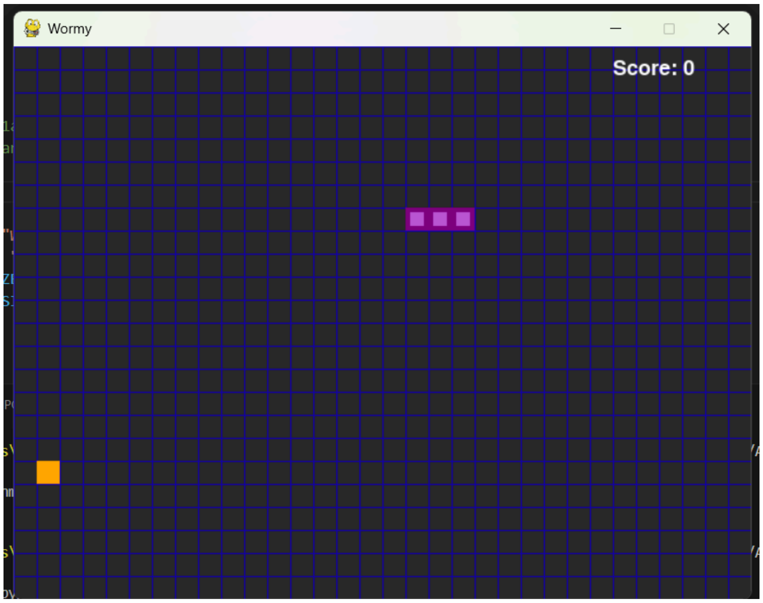
Draw the grid lines across the window so the game has a clear layout.

Output

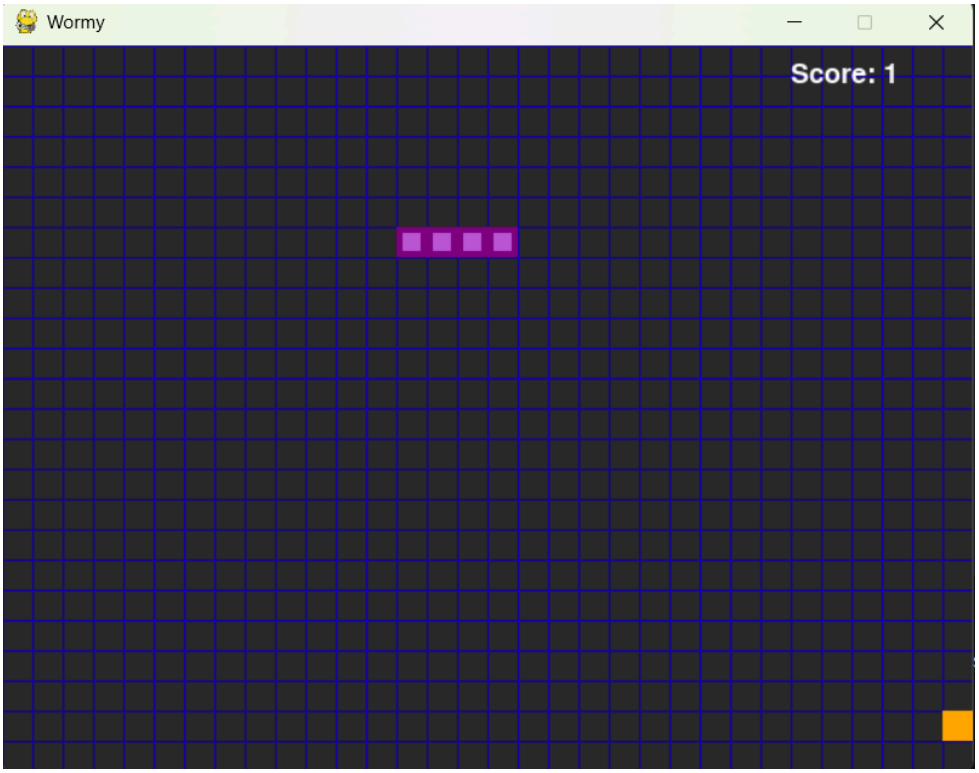


GAME OVER

Press R to play again | Q to quit



Score: 0



Score: 1