

Student n°: 200083757

Tile-Matching Game Report

13/12/2023

Contents

Tile-Matching Game Report	1
Introduction	3
Part 1- Planning	3
Part 2- Implementation	7
Structure	7
System Tests	7
Part 3- Conclusion	8

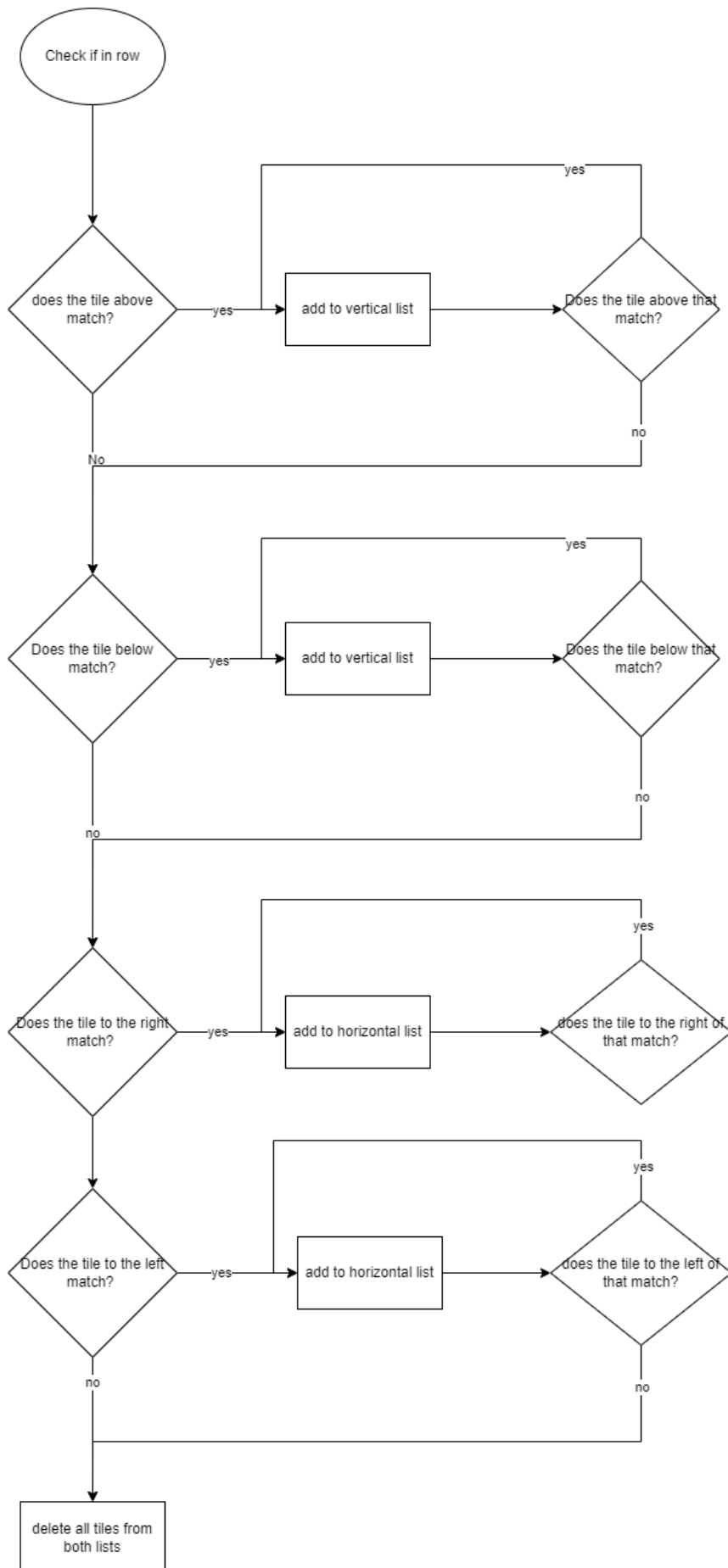
Introduction

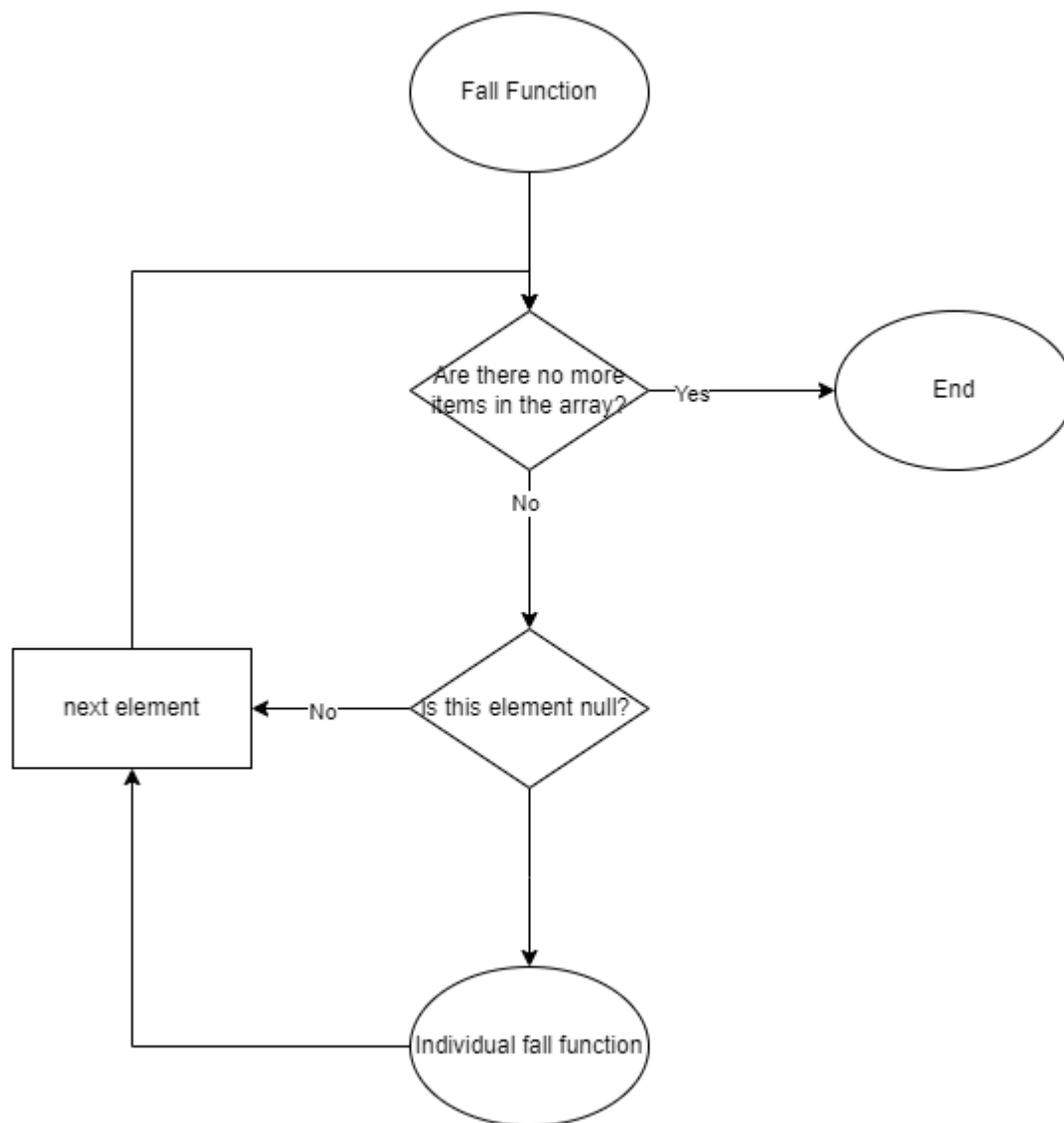
The aim of this assignment was to create a tile matching game where tiles could be swapped with each other to create sets of three or more and then disappear.

Part 1- Planning

My original aim for the project was to create a Use Case document and a Use Case Presentation, however I ran out of time. The Use Case document would have defined of all the possible Use Cases and the Presentation would have shown what they looked like.

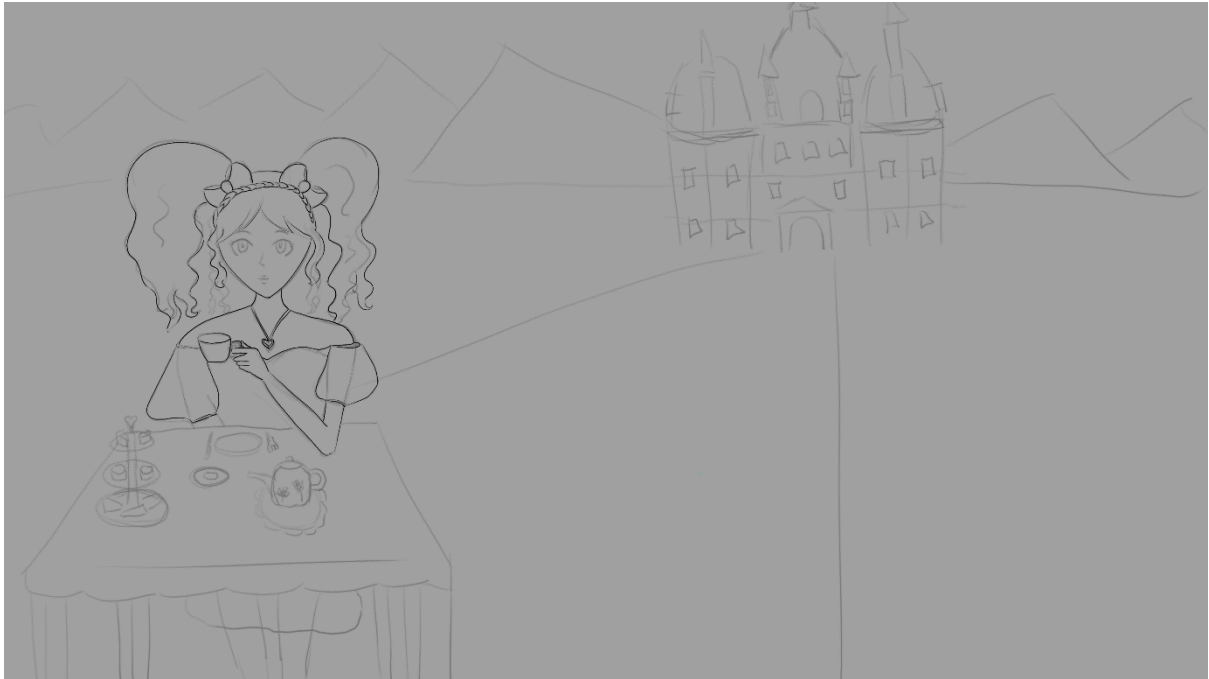
I was also planning on making a Design Specification with UML diagrams and Flowcharts to help plan my code. As it is I did create two flowcharts which are following:





The check-in-line function is inside of the checker class. This class was originally intended to have a "check if block" function as well, that would have extended the abstract checker class, but that didn't end up happening.

I was originally going to draw art for this project as well, and I have an example of the background here:



Part 2- Implementation

Structure

This project was created by using assemblies to help structure the code and enable easy use of Unit Testing. The project was also split into scenes: Menu and Game. There is an empty object in both the Menu scene and the Game scene called SceneManager that handles changing between the two. Inside of the Game scene, there is another empty called GameManager that handles the game logic. It contains the BoardManager Class that essentially brings all the code from the other scripts together. The jewels are defined as a prefab that the BoardManager class instantiates.

Inside of the assets folder there is also a levels folder that contains the csv files that defines the starting position for levels. Currently there is only a test level in there, but it hopefully it is enough for proof of concept.

System Tests

I used the Unity Unit Testing module to work on this project, creating unit tests for functions as I went. The fall function does not work in this project and therefore the fall test also does not pass. In fact, sometimes when run, it will crash the entire code. Every other test written so far will pass.

Part 3- Conclusion

I know that this project is very incomplete and is not at all done to a standard that I would normally feel comfortable uploading, however I thought it would be more sensible to upload something, regardless of how incomplete, than to upload nothing at all.

In an ideal scenario, I would have liked to complete the implementation and create 8 fully playable levels. I would have added more jewel types (through the Inspector), possibly in such a way that later levels had more jewels, making them harder. I wanted to create an endless mode where players could try and beat their high score.

Overall, I am not at all happy with this project and expect to get around 40 percent if that.