GADE6122_PART 2 -POE

Date – 08 October 2024

Teammates: Jet Kingston, Belinda Mtabane

Index:

Challenges in the POE – Belinda Mtabane...Page 3

Challenges in the POE – Jet Kingston..Page 4 - 5

Success in the POE Completion – Belinda Mtabane.........................Page 6

Reference page...Page 7 - 8

Belinda’s struggles

Challenges in the POE

I faced many difficulties in Questions 2 and 3, including issues with enemy attack and movement, despite the absence of errors. These seemed like logical errors, which were tough to untangle.

In Question 2.4, the Trigger movement method had to activate enemies after the hero successfully made two moves. My initial implementation followed the document's instructions, but during testing, none of the enemies were interactive.

I sought help from Sasquatch B Studios to break down C# code for enemy movement and spawning in random places (see A Better Way to Code Your Characters in Unity | Finite State Machine | Tutorial - Sasquatch B Studios, 2023). They provided great insights, but the logical error remained elusive and unsolved in a short span.

In Question 3, the hero needed to attack enemies, and enemies had to attack the hero. This was difficult due to the interaction issues with the enemies. However, implementing the logic to make the hero character detect adjacent tiles to exit and move to the next level was relatively easy. Yet, when I tried implementing this logic following McCaffrey (2024), the logical error in the code was hard to detect and resolve.

From the previous submission on the Grid game for the Level class within the code to ensure the functionality of our game, with guidance from (W3 Schools, 2024).

Thirdly, with this I was not able to make the health characters interactive within the gameplay.

Jet’s struggles

Challenges in the POE

The first issue I had with Part 2 was the GetMove method, which took an out parameter destination of type Tile and returned a Boolean indicating whether a valid move was found. The key component we used in the code was direction checking. With help from GeeksForGeeks (2022), we learned we needed two arrays: dx (x coordinate) and dy (y coordinate), representing movement in all four directions. GeekforGeeks (2022) also explained how to find all adjacent elements in a 2D array.

By applying this understanding, we could check all adjacent tiles. Next, we needed to research move selection, which involved choosing one random adjacent tile and updating the grunt’s destination. We used Microsoft's documentation (2024) to understand how to use a random generator and random.Next() to generate a random position for the grunt.

We implemented this by initializing a list of emptyTile to store valid move destinations. The code looped through the character's vision array, checking if each of the four direction coordinates was an empty tile.

It added found empty tiles to the list. After processing up, down, left, and right, it checked for valid moves. Once a valid move was found, it generated a random empty tile from the list and set it as the destination, returning true when a valid move was found.

Continuing with our POE, I struggled with creating an abstract method called ApplyEffect. With help from GeekforGeeks (2023), I realized my mistake. They helped me see that abstract methods are declared inside the abstract class only.

Problems raised in the POE

our grunt didn't spawn at the start of the game, our hero was unable to kill enemy grunts, movement was a bit off since instead of moving 1 tile at a time, the hero would move 4. My file full of all the code also disappeared during the POE which almost made me lose all my progress.

Conclusion

Therefore, with regards to part 1, part 2 was way easier to understand as most of the things in part 1 that we implemented came up again in part 2, which made things way easier to understand and figure out. Leaving only a few cracks in the walls that we needed to fill up our understanding of in order to sort out the issues.

Success in the POE Completion

With the tasks at hand, each team member was assigned questions that challenged our capabilities and techniques. In Question 2, the easier questions were completed promptly within a few days. This included making the gate appear from Part 1 of Gade, which had been an obstacle for me.

After many attempts at implementing it in the difficult route, I decided to follow the basics. This success made Gade seem more appealing and turned it into a challenge worthtackling, similar to Part 1.

In addition, for Question 3, my teammate and I collaborated to complete half of it due to its complexity, balancing our understanding and development. However, delving deeper into Question 3 took a toll on game progression, particularly with enemy attack and movement mechanics.

Question 4 was easier, as the document's complexity was broken down into the required steps. This process increased my knowledge in terms of communication and coding.

Reference

W3 Schools. 2024. C# Multidimensional Arrays,. [Online]. Available at: https://www.w3schools.com/CS/cs_arrays_multi.php [Accessed 08 September 2024].

A Better Way to Code Your Characters in Unity | Finite State Machine | Tutorial – Sasquatch B Studios.2023. YouTube video, added by Sasquatch B Studios. [Online]. Available at: https://www.youtube.com/watch?v=RQd44qSaqww [Accessed 07 October 2024].

McCaffrey, J. 2023. Data Anomaly Detection Using a Neural Autoencoder with C#. Visual Studio Magazine, 15 April 2024. [Online]. Available at: https://visualstudiomagazine.com/Articles/2024/04/15/data-anomaly-detection.aspx#:~:text=This%20article%20explains%20how%20to%20use%20a%20neural%20autoencoder%20implemented [Accessed 07 October 2024].

ArtCorVibes. 2024. Pixel Art Styles: The World of Pixels in 2024 84. [Online]. Available at: https://artcorvibes.com/pixel-art/#84 [Accessed 07 October 2024].

GeeksForGeeks. 2022. Move matrix elements in given direction and add elements with same value, 19 August 2022. [Online]. Available at: https://www.geeksforgeeks.org/move-matrix-elements-given-direction-add-elements-value/ [Accessed 08 October 2024].

GeeksForGeeks. 2022. Find all adjacent elements of given element in a 2D Array or Matrix, 15 November 2022. [Online]. Available at: https://www.geeksforgeeks.org/find-all-adjacent-elements-of-given-element-in-a-2d-array-or-Matrix/ [Accessed 08 October 2024].

Microsoft.2024. Random Next method implementation, 1 November 2024.[Online]. Available at: https://learn.microsoft.com/en-us/dotnet/api/system.random.next?view=net-8.0 [Accessed 08 October 2024].

GeeksForGeeks. 2022.Absract classes, 10 April 2023. [Online]. Available at:

https://www.geeksforgeeks.org/c-sharp-abstract-classes/

[Accessed 08 October 2024].