Test Plan

Section 1: Introduction

The wandering in the woods game is a basic simulation that shows two characters on a screen moving at random. The characters move around the screen until they come together or settle on the same spot. Two players and a predefined square screen playing field make up the basic wandering in the woods game. We've improved the game structure to offer more features and give users greater control over how they create the playing area and perform random simulations. The goal of our game is to allow students of various grade levels to participate in the wandering in the woods game at their own level of education and knowledge.

Our game is designed to be user-friendly for students of all ages, including teachers. We also wanted to appeal to a younger audience, so we went with a Oggy and Jack motif for the characters. The game is a pleasant educational tool that allows teachers and parents to teach children about mathematical functions. We designed the game so that kids may select their grade group, which then leads them to select extra aspects for their game or simulation. The user guide that is attached explains how youngsters can choose the game they wish to play in great detail. The general notion is that the game becomes more sophisticated as the children get older.

We tried to keep things as basic as possible for grades K-2. When children select the k-2 option, they will be given a square grid with only two characters in the upper right and lower left corners. We did provide this grade group the option of choosing between three distinct square grid sizes. The game will begin automatically after they select one of these grids, and the two characters will begin to move about the grid at random until they meet. The amount of moves it took the two characters to meet each other is indicated once they've met. We provided the possibility for students in grades 3-5 to construct their own grid.

The grid could be any size and shape, and it didn't have to be square. We also gave this group the option of selecting how many characters they want in the simulation. They had the option of selecting two, three, or four characters. The simulation starts and goes until the players meet each other after the users in this group customise the grid and the number of participants they wish. In a separate text file, the number of moves it takes for the players to meet is displayed and recorded. Users can run the simulation numerous times to find out how long it takes for characters to meet up on average, how long it takes for characters to meet up in the shortest run, and how long it takes for characters to meet up in the longest run. The 6-8 grade group is set up similarly to the 3-5 grade group, with the exception that users can now control players. Users can utilise this to assess how quickly the characters can meet up. It provides this grade group greater leeway than the others, allowing them to explore with new circumstances.

Section 2:

We had to construct a somewhat complex software to obtain the result for the game. The programme offers a variety of features and functions that enable us to achieve the objectives listed above. We had to test each part carefully and one at a time to ensure that everything functioned correctly. The items we tested that were critical to our program's completion are listed below:

2.1. Basic Setup of Wandering in the Woods Game

When the game's base gameplay was being developed, the first test was conducted. We had a screen open with an animated background, two characters starting at the top left and bottom right, music playing, and the option for users to control the two characters with

the wasd and arrow keys at this time. This test was performed to check that the audio, visual, and character movement were all working properly. We had to move each character around the screen while keeping the boundary in place to prevent them from escaping. This section of the programme was run roughly 5 times. Users were able to manipulate characters and hear sound, which was exactly what we wanted. We also created a draught of a user-acceptance test that walks people through the game. This test will determine the game's final structure.

2.2. Border Collision Sound and Character meetup

This test was performed check is sound plays when characters hit a wall and if the game stops when characters run into each other. We performed this test by manually moving the characters around the screen until they meet. This test also was used to make sure the step counter worked, and the total number of steps did appear when the characters met each other. We checked this function about 10 times to ensure the proper steps were tracked. We had no issues with this test, and the correct number of steps did appear.

2.3. Base Testing and Debugging

The purpose of this test was to see if sound plays when characters strike a wall and if the game pauses when characters collide. The characters were manually moved around the screen until they met in this test. This test was also used to ensure that the step counter functioned properly, and that the total number of steps appeared when the characters collided. We double-checked this function around ten times to make sure all the stages were recorded correctly. This test went without a hitch, and the correct number of steps appeared.

2.4. Different Map Sizes and Celebration Sound

More map sizes were generated, as well as new map backgrounds. When the characters came together, we added celebration noises as well. We wanted to make sure that the various map sizes worked with the code we already had, so we tested by shifting the characters about on all the various map sizes. On these various sized maps, we also tested the celebratory sound. Each map size was tried 5-6 times. They appeared to work as planned, as the process of adopting the additional map sizes was not dissimilar to the process of creating our initial map.

2.5. Menu

The goal of this test was to ensure that the menu functioned properly and that the buttons directed users to the relevant game types. Clicking each of the menu buttons and confirming that they led to the correct map size was done as part of the testing. We simply tested each menu button twice because there should be no issues with a button sending the user to another window. Each button functioned as expected.

2.6 Random Simulation

We had player movement, audio and visual components, various map sizes, and a counter added to the application at this stage. For the varied players and map sizes, we implemented a random simulation. To ensure that the random simulation worked properly, we ran through each of the game variants with varied player counts. This was a time-consuming exam because the larger maps took a long time to complete. Each map was tested three times, and the time it took for the characters to meet was lengthy. We were first concerned, but understood that with larger maps, the steps required for the characters

to meet should be considerable. The characters eventually met up, and the total steps were displayed.

2.7 Loop to Take User Back to Main Menu

We made it possible for players to return to the main menu at any time during the game. We wanted users to be able to go back a screen if they accidentally pressed the wrong button. We also wanted players to be able to return to the main menu and play the game again once it was ended. We verified this by testing each button, map size, and player count, as well as ensuring that users could return to the previous screen. We also ensured that the user could return to the menu after the simulation was completed. This feature worked perfectly for us and we had no issues with it.

2.8 Base Testing

This was a challenging day in general. We wanted to double-check everything and ensure that the functions and features we added were working properly. On the entire code, we ran roughly two dozen tests. We tried every option a person might choose and had no problems. We tweaked a few minor aspects, but the game appeared to be in good working order.

2.9 Statistics Test

The option to see the shortest, longest, and average runs of each game mode and player mode has been introduced. We next put the functionality to the test on each of the game modes a few times. The statistics were exported to a.txt file, which the user may see.

2.10 Final Testing

The goal of the test was to see how the entire code worked and how the data was displayed. The entire crew conducted their own tests to look for flaws or faults in the code. We discovered a few problems with the statistics and.txt files. We tweaked some code and conducted additional tests.