using System.Collections;

using System.Collections.Generic;

using UnityEngine;

using UnityEngine.SceneManagement;

public class Level1Spawn : MonoBehaviour {

int[] Shuffled = new int[100];

GameObject Block;

public GameObject enemy1;

public GameObject enemy2;

public GameObject enemy3;

public GameObject enemy4;

public int aliveenemy;

public DataController data;

public int killcount;

public bool levelend;

public int lifeRemaining = 10;

public int positionindex;

public int enemyindex;

public bool alive = false;

int framecount = 0;

GameObject enemy;

Vector3 position;

GameObject[] enemyArray = new GameObject[4];

Vector3[] positionArray = new Vector3[4];

public int[] Initarray = new int[4];

public int shot1 = 0;

public int shot2 = 0;

public float Compare;

int currentpos = 3;

int nextpos;

int swap;

bool assigned;

void Start () {

Block = GameObject.Find("Wall");

Block.GetComponent<Wallscript>().player = gameObject;

Shuffled = GetComponent<RandomWalk>().Shuffled;

data = FindObjectOfType<DataController>();

positionArray[0] = new Vector3(-4.5f, 1.15f, -4.5f);

positionArray[1] = new Vector3(4.5f, 1.15f, -4.5f);

positionArray[2] = new Vector3(4.5f, 1.15f, 4.5f);

positionArray[3] = new Vector3(-4.5f, 1.15f, 4.5f);

enemyArray[0] = enemy1;

enemyArray[1] = enemy2;

enemyArray[2] = enemy3;

enemyArray[3] = enemy4;

for(int i = 0; i < 4; i++)

{

Initarray[i] = 0;

} while (currentpos > 0)

{

nextpos = Random.Range(currentpos, 4);

swap = Initarray[currentpos];

Initarray[currentpos] = Initarray[nextpos];

Initarray[nextpos] = swap;

currentpos -= 1;

}

killcount = 20;

levelend = false;

}

void Update () {

if (levelend)

{

data.levelData.levelend = true;

data.playerData.LifeRemaining = GetComponent<playerhealth>().playerhealthnum;

SceneManager.LoadScene("Placement2");

}

else

{

if (!alive)

{

if (assigned)

{

if (Shuffled[shot1] <= 25)

{

enemyindex = 0;

shot1 += 1;

Compare = Mathf.Round(data.playerData.Prior[enemyindex].mean\*100f);

if (Compare >= Shuffled[shot2])

{

positionindex = Initarray[enemyindex];

}

else

{

positionindex = Mathf.RoundToInt(Shuffled[shot2] / 25);

if(positionindex == Initarray[enemyindex]&&positionindex<3)

{

positionindex += 1;

}else if(positionindex == Initarray[enemyindex] && positionindex == 3)

{

positionindex -= 1;

}

}

shot2 += 1;

}else if (Shuffled[shot1] > 25&& Shuffled[shot1] <= 50)

{

enemyindex = 1;

shot1 += 1;

Compare = Mathf.Round(data.playerData.Prior[enemyindex].mean \* 100f);

if (Compare >= Shuffled[shot2])

{

positionindex = Initarray[enemyindex];

}

else

{

positionindex = Mathf.RoundToInt(Shuffled[shot2] / 25);

if (positionindex == Initarray[enemyindex] && positionindex < 3)

{

positionindex += 1;

}

else if (positionindex == Initarray[enemyindex] && positionindex == 3)

{

positionindex -= 1;

}

}

shot2 += 1;

}

else if (Shuffled[shot1] > 50 && Shuffled[shot1] <= 75)

{

enemyindex = 2;

shot1 += 1;

Compare = Mathf.Round(data.playerData.Prior[enemyindex].mean \* 100f);

if (Compare >= Shuffled[shot2])

{

positionindex = Initarray[enemyindex];

}

else

{

positionindex = Mathf.RoundToInt(Shuffled[shot2] / 25);

if (positionindex == Initarray[enemyindex] && positionindex < 3)

{

positionindex += 1;

}

else if (positionindex == Initarray[enemyindex] && positionindex == 3)

{

positionindex -= 1;

}

}

shot2 += 1;

}

else if (Shuffled[shot1] > 75)

{

enemyindex = 3;

shot1 += 1;

Compare = Mathf.Round(data.playerData.Prior[enemyindex].mean \* 100f);

if (Compare >= Shuffled[shot2])

{

positionindex = Initarray[enemyindex];

}

else

{

positionindex = Mathf.RoundToInt(Shuffled[shot2] / 25);

if (positionindex == Initarray[enemyindex] && positionindex < 3)

{

positionindex += 1;

}

else if (positionindex == Initarray[enemyindex] && positionindex == 3)

{

positionindex -= 1;

}

}

shot2 += 1;

}

if (shot1 > 99)

{

shot1 = 0;

}else if (shot2 > 99)

{

shot2 = 0;

}

}

else

{

if (Shuffled[shot1] <= 25)

{

enemyindex = 0;

shot1 += 1;

positionindex = Initarray[enemyindex];

}

else if (Shuffled[shot1] > 25 && Shuffled[shot1] <= 50)

{

enemyindex = 1;

positionindex = Initarray[enemyindex];

shot1 += 1;

}

else if (Shuffled[shot1] > 50 && Shuffled[shot1] <= 75)

{

enemyindex = 2;

positionindex = Initarray[enemyindex];

shot1 += 1;

}

else if (Shuffled[shot1] > 75)

{

enemyindex = 3;

positionindex = Initarray[enemyindex];

shot1 += 1;

}

assigned = true;

if (shot1 > 99)

{

shot1 = 0;

}

}

framecount += 1;

if (framecount > 10)

{

var enemy = Instantiate(enemyArray[enemyindex], positionArray[positionindex], enemyArray[enemyindex].transform.rotation);

enemy.GetComponent<EnemyShot>().player = gameObject;

alive = true;

killcount -= 1;

aliveenemy = enemyindex;

if (killcount < 0)

{

levelend = true;

data.playerData.Level = 2;

}

}

}

else

{

framecount = 0;

}

}

}

}