AI For Games CW2

Ali Nouyan

Whilst trying to come up with the idea of what to do for this piece of coursework, I looked into Pac Man and saw that each of the ghosts in that game are scripted and behave differently which is what I will use as inspiration for my coursework. The basic idea was to be in an arena with 4 different enemies, you as the player will need to avoid being hit by these AI enemies and try to kill them. Each of the enemies are killed in different ways and also act differently. In this write up I will explain what each of the enemies will do.

1. The first AI Enemy (Red) will work similar to the weeping angels on Doctor Who. The player will have a light of some sort and whenever he faces the AI enemy or shines the light on him, he will stop moving. He will then carry on moving as the player looks away.

The way you will need to kill this first Enemy AI is by keeping the light on him for a certain amount of time. After the lights been on him for a few seconds he will dash towards the player, if he hits the player then the player will either die or lose a life.

1. The second AI Enemy (Blue) will be blind and move based on sound. Because of this he will stay stationary, listening out. When the player moves he will start moving towards the players position but if the player stays still then he will stop moving. There will be rocks scattered around the arena which the player can pick up and then throw. This AI will hear the rock and start moving towards its position. There will also be 3 boulders scattered around the arena. If the AI hears that he is close to the player, he will dash towards you in hopes of hitting you.

In order to kill this enemy you will need to throw the rocks in a way to make this Enemy AI run into the boulder, destroying the boulder and hurting himself. If you do this will all 3 boulders then this enemy will be dead.

1. The third enemy AI (Green) will also move based on the players movement. This enemy will track the Players click and keep count of them. Every third click this enemy will rush to the position clicked, forcing the player to move once again. If the player doesn’t click on the screen for a set amount of time then this Enemy AI will start to get angry and attack the player. The first attack won’t really be much of an attack but, this AI will just go to the player’s position. The second attack will be a dash and the third attack, he will throw a rock at the player, wait for the rock to reach the players position then dash towards the player.

In order to kill this AI the player will need to lure out its attacks because after the third attack, the rock and dash, he will then do it again for the 4th attack but he’ll warn himself out too much and die.

1. For the fourth AI (Yellow) I want to introduce fuzzy logic, for the other enemies I will just use the normal method to switch from states. I will use variables such as Anger, Distance to player, Distance to the most recent rock thrown, Click Count, the flash light somehow, health, etc. I may change the variables I use depending on what’ll work best whilst imp actually creating this AI, I’m going to have the FSM for this enemy AI be slightly simpler due to the fact that vie never used fuzzy logic before. The states will include, Chase, run away, throw rock, dash and death.

The way you kill this enemy AI is by killing all of the other enemies first, in a way this enemy AI is the boss. Once you kill all the other enemies, you will need to get this enemy to dash towards you and then once he finishes the dash he will die. I want the player to have to lure out the attack instead of killing him off as soon as the last enemy died because it could get confusing for the player is 2 enemies die at the same time at the end of the game.

Whilst coming up with the ideas for my enemies, I wanted a lot of their movements and attacks to be similar but the enemies will be different based on the requirements for them to attack and just simply the way their FSM is setup. For example all the enemies have a dash but the first AI’s dash will be triggered from the light being on him for too long whereas the second one will dash as soon as he gets close to the player.

Conclusion:

Now that I’ve finished the demo, I feel like everything went as smoothly as coding goes, all of the AI enemies work well together and don’t really get in each other’s way. One thing I didn’t take into account whilst coming up with this ideas is when you’re trying to play the game with all 4 AI Enemies it gets cluttered and it becomes slightly harder to tell which enemy is which, and this is coming from the person who made this. If someone were to pick this up and play it with not knowing anything about the game it can get confusing.

Although with this being said I could easily get around this by adding a start screen where the player will have the option to view the enemies and what they do before they actually get into it. There are lots of things I would add to this to just simply improve the gameplay experience such as being able to choose which enemies you want to face in the arena by just having a check box before the game starts. Other things I would add if I had more time are small things such as sounds cues to let the player know when somethings about to happen. E.G I could have a little build up sound effect right before the dash or have a death sound when an enemy dies.

One more thing I decided not to add for the submission is health for the player. Whilst testing this I realised it can be difficult and I didn’t want to have to constantly restart and repeat it over and over to test one thing because I keep dying to I left it out for now.

In conclusion I enjoyed this project and had fun using different AI techniques to create this demo and just seeing it all come together from having just one enemy that works to all 4 together. I’ve also learnt quite a bit whilst making this demo, especially fuzzy logic as I hadn’t really practised it at all before I tried to use it in this. I also added an average function on the end of my fuzzy script as I thought I would come in handy.