Spike: 7
Title: Emergent Group Behaviour

Author: Thomas Latimer, 9718648

Goals / deliverables:

The primary goal of this spike is to create some additional A.I. Behaviour such as adding Cohesion, Separation and Alignment. In some cases, as provided in the instructions, some blanks need to be filled in. Other functions are updated in order to use the additional steering behaviour in the program. What we want to do is get these behaviours to work together and declared in one function instead of trying to get them to work separately. Also they need to be adjustable.

Technologies, Tools, and Resources used:

- PyCharm 3.1. for Microsoft computers
- Python 3.6, for separate testing.
- Additional research e.g. the python website.
- Microsoft word for the construction of this document.
- Refer to simple code A.I. in lab 6.
- This spike is known as Spike_7_report_9718648.

Tasks undertaken:

- Read through the criteria and go over the lab 6 code
- Created a definition for Cohesion, separation and Alignment.
- Enable some wondering behaviour in the flocking function so it can work with the other steering forces.
- Created a weighted some in the flocking function so that the steering behaviours become adjustable.
- Make sure that when you adjust the steering behaviours that the program still runs while doing while the effects take place.
- Keep testing to make sure the program works and find out how it can possibly be extended.
- Run code.
- Write Spike Report.

What we found out:

The outcome was testing the program, Making sure the errors were corrected and what happened once the code was executed. I wanted to make sure each of the functions for cohesion, alignment and separation would work separately so that when they do we can try to combine the behaviours together. I didn’t know that we needed to create a function so that they could work together then create a function referring to those behaviours with the weighted sum as well. I assigned keys to each adjustable steering behaviour that we needed to test out such as ‘Q’ and ‘A’ for Wander, ‘W’ and ‘S’ for Alignment, ‘E’ and ‘D’ for Separation and ‘R’ and ‘F’ for Cohesion. It took some testing but I managed to create a weighted sum for each steering behaviour and give each of them a set value because some behave different depending on how much each weighted sum has. I didn’t have a set value for ‘Wander’ I just left it at 0 because if you increase it just wanders around just by pressing a key. I made the ‘Alignment’ weight 10 because I noticed when I reached this point all the agents starting moving in the same direction and the occasional get stuck, then eventually get their act together which was common during testing. ‘Separation had a weight of 170. It took a while to see some sort of reaction for separation because we wanted them to react to each of the agents present on the screen but at this weight it was able to show results by actually staying away from each other the best they could. ‘Cohesion’ was 50 because we wanted them to stick together as a group and this weight was the most effective. We made sure to test them separately instead of a combination of them to confirm that each function worked properly.

Open issues/risks:

- I did have some trouble with the code prior because even though the comments specified what needed to be done with the program such as fill in some of the missing blanks but the issues I had was how to fill in those missing pieces.
- I didn’t know that I needed to created extra variables elsewhere because I thought that we needed to create a function for each steering behaviour, find a way for them to work together and use a weighted sum to make the behaviours work at the press of a button.
- You had to figure out how to link or create relationships between each of the functions. But at first it was tricky to.
- Make sure to remember which key does what when adjusting the steering behaviour or you’ll have no idea what you are testing or changing.

Recommendations:

- Make sure you use one program you are comfortable with and stick with it because performance varies on each program you come across when coding in them.
- Remember what keys adjust what behaviour so you can keep track of what you do.