Flocking Dead

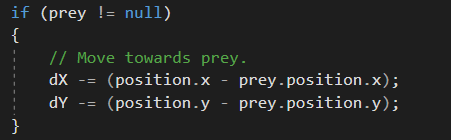
Beau Wijkstra, Quintin Yu.

500803785, …

# Opdracht 1:

## Finish the Hunt function in the agent class, such that zombies will chase after other agents.

Om de zombie naar de prey te laten bewegen, moet er bij de dX en dY een getal worden toegevoegd dat de positie van de prey vergelijkt met de positie van de Agent. Dit doen we letterlijk met een vergelijking. We doen de positie van de Agent – de positie van de prey:



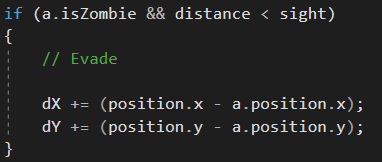
Het resultaat wordt dan dus dat de Agent naar zijn prey toegaat.



# Opdracht 2:

## Finish the Evade code - inside the Flock( ) function - in such a way that agents will try to evade zombies.

De evade code is eigenlijk heel erg vegelijkbaar met de hunt code. Tenminste dat was onze gedachtegang, dus dat hebben wij ook geprobeerd. Dezelfde code alleen dan omgekeerd.



En het resultaat is dan uiteindelijk dat de “Prey” dus de zombies ontwijkt.

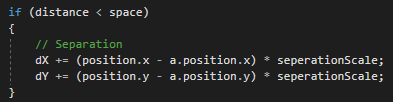


# Opdracht 3:

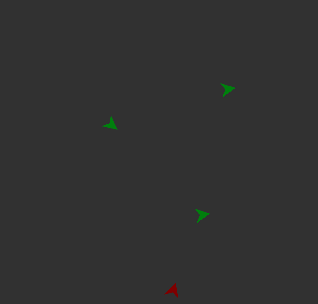
## Add a parameter to the separation code inside Flock(), in order to scale dX and dY.



Eerst hebben we een simpele scale toegevoegd. Doormiddel van deze scale veranderen we de richting waar een entiteit naar toe gaat wanneer hij een of meer non-zombie entiteiten tegen komt.



Dit hebben wij toegepast op de separatie code voor de non-zombie entiteiten. Door de scale te verhogen of te verlagen kunnen wij de richting van de entiteit makkelijk beïnvloeden.



Hier is bijvoorbeeld te zien dat de non-zombie entiteit rechts boven weg gedraaid is van de andere twee en gelijk weg loopt. Dit doet hij omdat hij weg wilt van de andere non-zombie entiteiten. Hierbij word de Vector (position.x – a.position.x) vermenigvuldigd met de scale. Door deze vermenigvuldiging word de desired position verplaatst.

# Opdracht 4:

## Explain why separation is important in flocking.

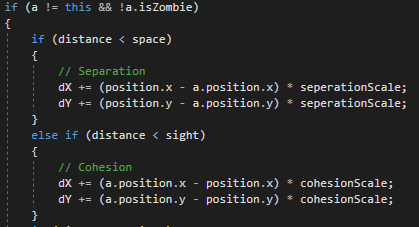
Separation is belangrijk in flocking om er voor te zorgen dat de entiteiten niet om elkaar gaan staan. Wanneer ze op elkaar staan en een zombie zou in de buurt komen heb je namelijk de kans dat alle non-zombie entiteiten tegelijk veranderd worden in zombies.

# Opdracht 5:

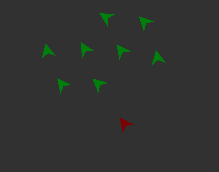
## Implement cohesion. Use a parameter to scale dX and dY.



Voor de cohesion heb ik een nieuwe scale aan gemaakt genaamd cohesionScale.



Wanneer de space groter is dan de distance maar distance kleiner is dan sight zullen de non-zombie entiteiten samen gaan lopen. Op deze manier zullen ze groepen gaan vormen en via deze groepen gaan bewegen.



Hier is bijvoorbeeld te zien dat ze in een groep bewegen. Ze proberen hier met de hele groep van de zombie weg te rennen.

# Opdracht 6:

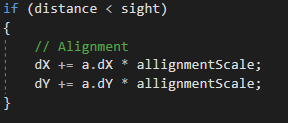
## Explain why cohesion is important.

Cohesion is belangrijk in flocking zodat de non-zombies gaan bewegen richting de gemiddelde positie van de groep. Op deze manier lopen ze de zelfde richting in.

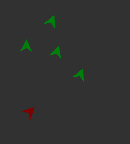
# Opdracht 7:

## Implement alignment. Use a parameter to scale dX and dY.

Alignment is een van de meest straight forward, je moet er gewoon voor zorgen dat de dX en dY zich aanpast aan dat van de andere agents. Dus dan krijg je iets als dit:



En dan kijken alle agents dus naar dezelfde plek als ze wegrennen van een zombie:”



# Opdracht 8:

## Explain why alignment is important for flocking.

Alignment is belangrijk omdat we willen dat de agents dezelfde kant op gaan, als je alleen cohesion en seperation hebt dan gaan ze de hele tijd naar elkaar toe en op elkaar af, en dan zitten ze in een soort loop waar ze niet uit kunnen, door alignment gaan ze een richting op met z’n allen.

# Opdracht 9:

## Using trial and error testing, find out what parameter values for separation, cohesion and alignment work well to create the following scenario’s

## 

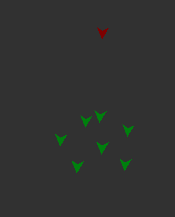
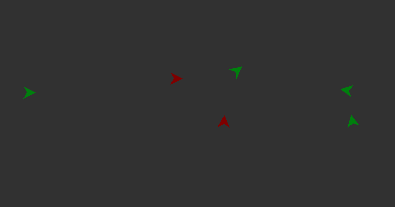
## a) Agents crowd together in fear of the zombies

## b) Agents spread out, making it harder for the zombies to catch them

## c) A single agent is trying to outrun 14 very slow zombies

## 

## If necessary, change the speed of your zombies to make them faster or slower. Explain (for each scenario) your choice of parameters using screenshots, and describe how you achieved the desired effects.

1. Voor dit scenario willen we dat de agents weinig Seperation hebben en veel cohesion, zodat ze naar elkaar toe gaan.
   1. We hebben dus de CohesionScale omhoog gezet en de SeperationScale omlaag gezet. Om precies te zijn hebben we de Seperation veranderd van 2.5 naar 0.5 en de cohesion van 0.01 naar 0.1.  
        
      Dit is dan het resultaat.
2. Om de agents te verspreiden moet je het omgekeerde doen van a). De seperation moet omhoog en de cohesion moet omlaag. Ook hebben we de allignment omlaag gezet.
   1. 
3. Ik heb alle parameters op nul gezet en de horde wat omhoog gezet, nu is er 1 agent die door erg veel zombies achtervolgd
   1. 