Programmeren opdracht 3: Intro to agents and agent-tools - individueel

## De tutorial

Het grootste voordeel van Unity is dat het erg toegankelijk is voor nieuwe gebruikers door de duidelijke interface. Het is makkelijk te leren maar er zijn ook vele mogelijkheden waardoor het voor gevorderde gebruikers nog steeds geschikt is. Voor mij het nadeel was dat bij het coderen juist wel heel veel voorkennis van Unity nodig is om goed gebruik te maken van alle voor gedefinieerde instanties en objecten. Het is super uitgebreid doordat al erg veel gemaakt is maar daardoor is het voor een nieuwe gebruiker moeilijker om te begrijpen ten opzichte van een andere taal.

Ik heb super veel tijd gestopt in het proberen uit te breiden maar dit is mij naast dat wat ik heb op dit moment niet gelukt. De interface alleen was al best veel om uit te vogelen en de code ernaast viel mij heel erg zwaar. Ik heb alle 2 de simulaties gemaakt en ook nog extra tutorials geprobeerd. Ik lever beide de tutorials vanuit canvas in en daar zal ik verder in dit document naar refereren.

## Stateful agent

De simulatie die ik heb proberen te maken is tikkertje. Er is dus een agent die probeert tegen de enemies aan te lopen.

De agent kan is meerdere states zijn. Een agent kan op zoek zijn naar enemies of achter een enemy aan rennen of ze kunnen, terwijl ze achter een enemy aan rennen, andere agents roepen om te komen helpen. De initiële staat van de agent bestaat dus uit hun positie en waar ze naartoe lopen. De initiële staat is dus de state waar een agent in zit als hij/zij inspawned.

De functie See neemt de state van de omgeving dus waar obstakels zich bevinden, of er andere agents in de buurt zijn en luistert naar geluid van een andere agent. Ook kan die enemies zien. Vervolgens gooi de See functie het gepaste precept eruit. Dus bijvoorbeeld bij het horen van andere agent zal deze worden waargenomen en gaat de agent naar het geluid totdat hij op locatie zijn of hij een ander enemy ziet.

Vervolgens bepaalt de functie Act wat er daadwerkelijk gebeuren moet. Bijvoorbeeld als er een geluid gehoord wordt, dan wordt de A of actie het volgen van het geluid. Of als de agent een enemy ziet, dan gaat de agent daar achteraan.

## Dichotomiën

1. In mijn simulatie is