

# Battleship

Gijs Veerman

Studentnummer: 1829579



# InhoudsOpgave

<b>InhoudsOpgave</b>	<b>1</b>
<b>ProbleemBeschrijving</b>	<b>2</b>
<b>Eisen</b>	<b>2</b>
<b>Algoritme</b>	<b>3</b>
<b>Gemaakte keuzes</b>	<b>5</b>

# ProbleemBeschrijving

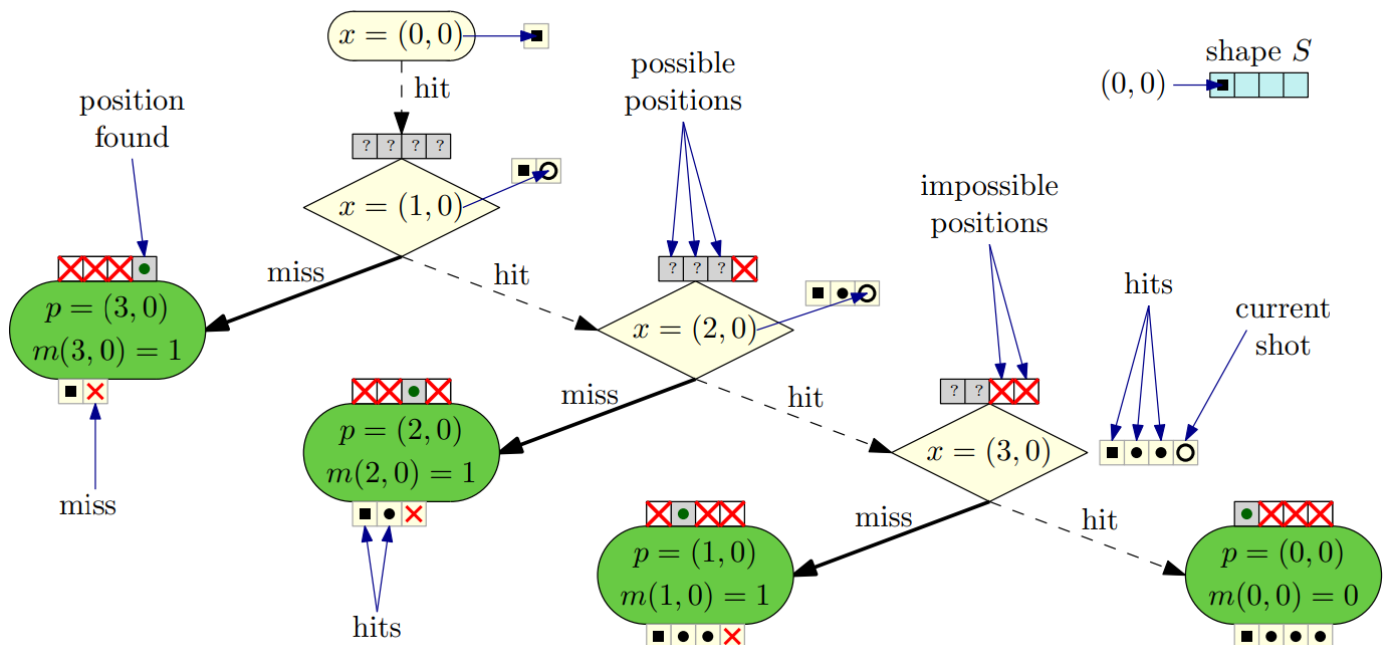
Verpleeghuis Vivium vroeg aan me of ik een bordspel om kon zetten in een één speler computerspel. Door een algoritme te maken voor het bordspel kunnen de bewoners in het verpleeghuis toch nog leuke bordspellen spelen zonder dat er een andere speler nodig is. We hebben het er even over gehad en zijn uiteindelijk gekomen op Zeeslag. Een leuk spel waar je de locatie van iemand anders zijn boten moet gokken. Je kan zelf je eigen boten plaatsen waardoor er een beetje strategie bij zit. Een boot zinkt als je de locatie van de hele boot hebt gegokt. De eerste die alle boten heeft laten zinken is de winnaar. Ze hebben ook aan me gevraagd of ik wat unieks in het spel wil doen zodat het nog een leuke uitdaging geeft.

## Eisen

- Het moet makkelijk speelbaar zijn. Dus iedereen moet weten hoe het werkt en wat de opties in het spel zijn.
- Het algoritme moet te verslaan zijn. Als iemand de hele tijd verliest wordt het spel minder leuk waardoor ze er minder gebruik van gaan maken.
- Er moet iets unieks aan zitten. Hierdoor wordt het spel nieuwer en de strategieën wat interessanter.
- Het moet goed te overzien zijn. Er moet niet te veel gebeuren want daardoor kan je heel makkelijk het overzicht kwijt zijn.
- Het moet makkelijk interactief zijn. Maak het plaatsen van de schepen en de gokken makkelijk. Hierdoor kan je je meer bezighouden met de strategie die je wil toepassen.

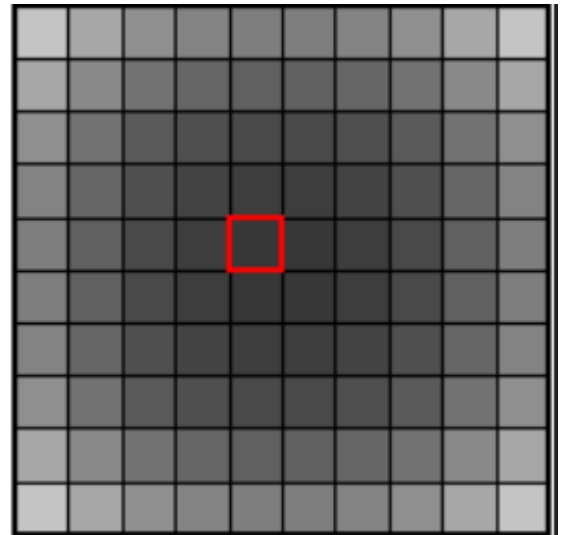
# Algoritme

Het algoritme dat ik heb gekozen is gebaseerd op simpele vormen. Dus alleen maar vierhoeken. Het werkt als volgt. Het moment dat je raakt gaat het in jaag modus. Het kijkt eerst naar alle mogelijkheden vanuit de hit die je hebt gemaakt. Hij kijkt daarna naar wat de beste mogelijkheid zou zijn welke kant je op kan. Hierdoor kijkt het simpelweg naar waar de meeste ruimte is. Hier is de meeste kans dat er een schip staat. Vervolgens maakt het de eerste gok in die directie. Als het een andere hit is gaat het vervolgens verder die kant op totdat het mist. Als het schip dan nog niet is gezonken gaat het de andere kant op. Dus als het eerst de hele tijd naar rechts gokt en het mist. Dan gaat het door naar links. Als het vervolgens mist gaat het verder naar onder en boven en doet het hetzelfde totdat er een schip gezonken is. Op het moment dat er een schip gezonken is, gaat hij verder over op zoek modes.

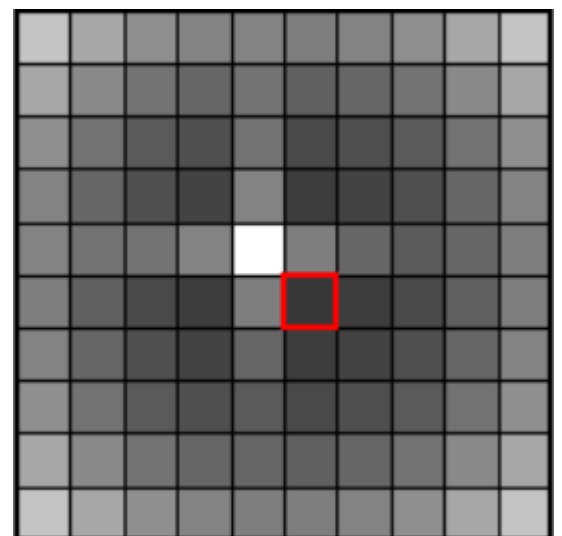


De zoekmodus kijkt naar de lengte van het langste schip dat er nog is. Vervolgens kijkt het naar waar dat schip mogelijk geplaatst zou kunnen worden. Daarna pakt het de coördinaat die het meest voorkomt in alle mogelijke posities. Stel dat er acht plekken zijn waar het schip zou kunnen worden geplaatst. Dan kijkt het naar welke coördinatie het meeste overeenkomt. Als er meerdere coördinaten zijn die evenveel overeenkomen, pakt het een van die coördinaten. Het moment dat het dan raakt gaat het over in jaag modus en anders blijft het een zoekmodus en gokt het ergens anders.

Elke keer kijkt het dus naar wat de beste opties zijn. Hoe donkerder hoe beter in dit geval. Hier doet het dus een gok in het midden.



Als het mist verandert het patroon zoals hier rechts. Dan berekend het weer de beste opties en gokt het weer de beste optie.



## References

*Battleship*. (n.d.). DataGenetics. <http://www.datagenetics.com/blog/december32011/>

Crombez, L., Fonseca, G. D. d., & Gerard, Y. (n.d.). *Efficient Algorithms for*

*Battleship*. [https://pageperso.lis-lab.fr/guilherme.fonseca/battleship\\_conf.pdf](https://pageperso.lis-lab.fr/guilherme.fonseca/battleship_conf.pdf)

## Gemaakte keuzes

Ik heb gekozen voor dit algoritme, omdat dat het meest toepasbaar was op de situatie die ik wilde creëren. Vivium en ik wilden een algoritme dat te verslaan was, maar ook weer niet te makkelijk. Hierdoor heb ik gekozen om voor de jaag methode de paper te pakken en dat algoritme een beetje aan te passen zodat het ook op verticale schepen kan worden toegepast. Hierdoor heeft het algoritme op zijn max drie gemiste schoten per schip. Door dit vervolgens te combineren met de zoekmethode van de blog, krijg je een leuk algoritme om tegen te spelen. We wilden het spel ook uniek hebben, ik heb daarom mijnen toegevoegd. Deze mijnen plaats je op het moment dat je de schepen plaatst. Het moment dat je de op een mijn gokt moet je een beurt overslaan. Hierdoor gaat de speler extra goed nadenken over waar hij zijn schepen en mijnen plaatst.