**Baseline**

*Jasmijn – Sabrina – Duco*

***Ons doel:***

Om te beginnen willen we ons doel definiëren: in zo’n kort mogelijke tijd tot de beste oplossing van Rush Hour komen met de minst mogelijke stappen. In ons geval gaat dit over stappen en geen moves, dus een auto beweegt altijd maar één vakje per keer.

***Wat hebben we al gedaan:***

We hebben al best veel tijd in de visualisatie gestopt, dit is praktisch af. Daarnaast hebben we een solver geïmplementeerd met een methode om meerdere keren een algoritme te runnen en te importeren via de CLI (command line interface). Je kan daarbij kiezen welke solver je wilt gebruiken, hoe vaak je het spel wilt laten oplossen, of je elke stap wilt visualiseren of niet en het geeft na het oplossen de statistieken van het oplossingsproces weer. Verder heeft game.py (de klasse Game en het script om de game via de CLI te spelen) de mogelijkheid om te kiezen welke game je wilt spelen. Bovendien hebben we een breadth solver en een random solver geïmplementeerd, bij de random solver hebben we het zo geïmplementeerd dat de oplossing per solve steeds beter wordt en dat er geen repetitions worden uitgevoerd. Als laatst is de README ook helemaal bijgewerkt.

***Hoe gaan we nu verder:***

We willen graag verder vanuit het idee van een breadth solver, het belangrijkste waar we daarbij op gaan focussen is het toevoegen van heuristieken en daarmee experimenteren.

*Het aantal stappen is per oplossing altijd hetzelfde, omdat de breadth solver altijd de beste oplossing opzoekt, de tijd om het spel op te lossen is vooral het onderdeel waar de heuristieken invloed op zullen hebben.*

Hier de breadth solver die game 1 10 keer oplost:

Afbeelding met tekst, schermopname, Lettertype

Automatisch gegenereerde beschrijving

Hier de breadth solver die game 3 10 keer oplost:

Afbeelding met tekst, schermopname, Lettertype, ontwerp

Automatisch gegenereerde beschrijving

Beiden spellen hebben een bord van 6 x 6, maar het 10 keer oplossen van bord 3 duurt al 31 keer zo lang als het 10 keer oplossen van bord 1.

Verder duurt het moeilijkste bord van 6 x 6 3.8 seconden om op te lossen en het eerste 9 x 9 bord duurt 399 seconden om op te lossen. De tijd neemt dus exponentieel toe. Om deze reden zullen we heuristieken moeten toepassen om ervoor te zorgen dat de tijd zo min mogelijk zal toenemen.

Het aantal seconden is slechts een indicatie, we streven ernaar om het aantal borden dat is bezocht en het aantal keren dat een bord is bezocht zo min mogelijk te maken.