Computational intelligentie Sudoku

**Planning:**

* **Plan van aanpak**
* **Verdiepen in algoritme**
* **Implementatie**
* **Testen**
* **Verslag**
* **10 binnen hengelen**

**Plan van aanpak**

* Class sudoku object
* Class minisudoku object
* 2D array voor rijen en kolommen met 81 elementen
* Random getallen invullen kijken welke ontbreken
* Hill climb algoritme

**Probleem definiëren:**

* Begintoestand -> Input van user geparsed in 2D array
* Hill climbing algoritme die wordt aangeroepen door heuristische functie en begin toestand
* Eindtoestand -> 2D array zonder nullen die voldoet aan sudoku

**Hill climb algoritme:**

* Willekeurig kiezen we een minisudoku object.
* Als die twee getallen of meer heeft die 0 nul zijn:
* - Proberen alle mogelijke swaps.
* - Daarvan heuristische waarde berekenen en kleinste pakken.
* Weer naar stap 1

**evaluatiefunctie**

* tellen we voor iedere horizontale rij en verticale rij het aantal dubbele getallen.
* Kiezen we toestand met laagste evaluatie functie en die vullen we in.

**Eisen:**

* Makkelijk grid kunnen kiezen en daar simpel verwisselingen in maken
* In kolommen en rijen apart kunnen aanpassen hoeveel dubbele getallen erin staan
* Weten welke getallen vast staan.