**Plan van aanpak**

Er zijn in de opdracht een aantal constraints meegegeven waarmee we moeten werken. Dit zorgt ervoor dat het aantal mogelijkheden groot, maar niet oneindig is.

Ten eerste moet 20% van het oppervlak(4800 m2) bestaan uit water. Dit water moet een hoogte/breedte verhouding hebben van tussen de 0.25 en 4. Er mogen maximaal 4 aparte stukken water geplaatst worden.

Hiernaast kunnen er 20, 40, of 60 huizen geplaatst worden. Binnen deze huizen zijn drie types te onderscheiden. Het percentage van elk type huis van het totaal ligt vast. Ook de grootte van de huizen en de hoeveelheid minimale vrije ruimte die ze hebben liggen vast.   
Binnen deze constraints is het aan ons om een optimale indeling te vinden. We kiezen ervoor om hierbij eerst het water te plaatsen. Dit doen we niet geheel random, maar aan de hand van bepaalde, door ons bedachte, constraints. Ook kijken we bij elk stuk water eerst of het niet overlapt met een ander stuk water. Het gaat hier dus om een constructief algorithme, en niet om een iteratief algorithme. Hieronder zal ik uitleggen hoe we dit probleem aanpakken, zowel met betrekking tot de achterliggende theorie, als de code.

Het eerste dat we doen in ons programma, is het water in de kaart plaatsen. Dit gebeurt niet geheel willekeurig. We hebben namelijk een aantal constraints aan het water meegegeven. De reden hiervoor is dat wanneer het water compleet willekeurig wordt geplaatst, er te veel situaties voorkomen waarin het water overlapt met andere stukken water of met de rand van het gebied. Dit zorgt ervoor dat het programma extreem langzaam wordt, omdat het water plaatsen steeds opnieuw moet gebeuren totdat het toevallig een keer goed gaat.

De manier waarop we het water plaatsen is als volgt: we geven van te voren zelf op (hard code) hoeveel stukken water we willen gaan plaatsen (1, 2, 3 of 4). Vervolgens genereren we voor elk stuk water een willekeurige linkeronderhoek, een willekeurig oppervlak, en een willekeurige hoogtebreedteverhouding. Deze waardes worden gegenereerd binnen bepaalde kenmerken, die we zelf opgeven. Deze grenzen hebben te maken met de eisen van de opdracht (bijv. het wateroppervlak is maximaal 4800 m2) en met het eerdergenoemde probleem van dingen compleet random doen: de kans dat er dan overlap ontstaat is te groot. De oppervlakte en de hoogtebreedteverhouding zijn samen genoeg om de hoogte en de breedte te berekenen. Vervolgens wordt gecheckt of het gemaakte oppervlak niet overlapt met ander stukken water of met de rand.   
In python gaat dit allemaal als volgt: in de main functie wordt een aantal stukken water opgegeven. Voor dit aantal wordt in een while loop steeds een willekeurige linkeronderhoek gegenereerd, en water aangemaakt.   
Het water heeft een eigen class, die in de main functie wordt aangeroepen. De belangrijkste functies in deze class zijn placewater en distancewater. Placewater genereert de verhouding en het oppervlak, en plaatst het water. Het water wordt alleen geplaatst wanneer de het gegenereerde oppervlak niet over de rand van het gebied gaat en het nieuwe totale wateroppervlak kleiner is dan het totaal te plaatsen oppervlak.

Vervolgens kijkt distancewater of het geplaatste stuk water overlapt met andere stukken water. Wanneer dit het geval is returnt deze functie False en begint de loop in main (die de linkeronderhoek genereert en water aanroept) opnieuw.

Hierna is het van belang de huizen in een bepaalde indeling in de kaart te plaatsen, en te kijken wat de waarde is van deze indeling. Om dit te doen, hebben we eerst een class in python geschreven genaamd house. In deze class staan een aantal kenmerken van de verschillende types huizen. Ook staan er een aantal functies in, die bepaalde gegevens kunnen returnen. Het gaat hier om de coordinaten van de linkeronderhoek van elk huis, de afstand tot het dichtstbijzijnde huis, en de vrije ruimte rond het huis. De functie distance berekent de afstand tussen het te plaatsen huis en alle andere huizen. Als de laagste uitkomst van distance onder de 0 ligt, kan het huis niet geplaatst worden: dan is er immers overlpa met een ander huis.

In de main functie ligt vast hoeveel er van elk type huis moeten worden geplaatst.

Als dit allemaal gelukt is wordt de waarde van de indeling berekend. Dit wordt gedaan door middel van de uitkomst van distance. De waarde van een huis wordt immers beinvloed dor de