

# DE PIXELATOR

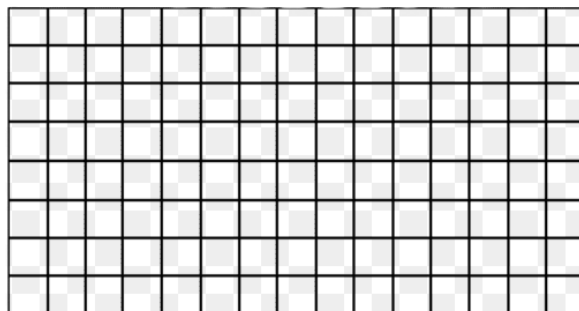
*Deze uitvinding is een metalen doos die, met de hulp van binaire stelsels, een tekening in een block-tekening kan veranderen, voor videospelkarakterontwerpers die niet in blokken kunnen tekenen.*

## HOE HET WERKT

*Er zijn vier stages en elke is heel belangrijk maar ook zo simpel dat het waanzinnig is.*

### LEVEL EEN: DE INPUT

*Iedereen weet dat geen programma's, computers of machines werken zonder input. Ook AI zoals Gemini kan niks als je het geen input geeft. (nou, het kan wel iets randoms genereren, als je de code een beetje verandert, maar dat helpt je natuurlijk niet.) En de input hier is een tekening. Geen blocky tekening maar een normale. Dan wordt het gescanned in een heel speciale manier. Het "scanner" is eigenlijk 15 of 20 (hoeveel je blokken op de y-axis wilt) en dan gaat het op een grid zoals dit:*



De camera gaat perfect met elke camera elke plek scannen, en als het kleur ziet – ook een klein vlekje dan voert het in als 1 en als er geen kleur is den voert in als 0. Aan het einde ziet het eruit als dit:

```
000001111111111100000
00011100000000111000
00111011000011011100
00110011000011001100
01110000000000001110
00110110000001101100
001110111111111011100
00011100000000111000
000001111111111100000
```

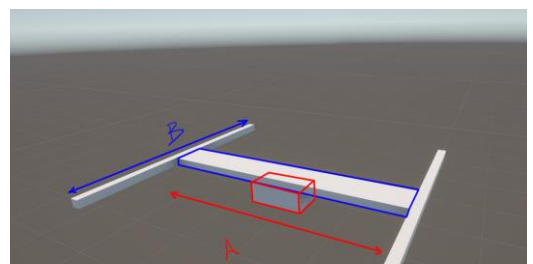
(Het echte is eigenlijk meer gedetailleerd maar dat kan ik niet laten zien. Ook zijn er geen kleuren; dat heb ik gewoon toegevoegd om het beter te laten zien.). Dan wordt dit binaire tekening in een heel lange lijn zoals dit:

```
000001111111111100000000111000000001110000011101100001101110000110011
000011001100011100000000000011100011011000000110110000111011111111101
1100000111000000001110000000111111111100000
```

Dan wordt deze informatie vervoerd naar:

## LEVEL TWEE: DE BREIN

Wat je eerst moet snappen is dat de brein niet letterlijk een brein is – er is niet een minicomputer of tenminste een raspberry pi. Het is gewoon een simpele programma dat zelfs ik, een elfjarige jongen, zou kunnen coderen. De brein (ik noem het gewoon de brein want dat is denk ik makkelijker dan de geprogrammeerde code-algoritme) is direct aangesloten aan een arm zoals op een van die klauwmachines waar je nooit iets wint. Ik bedoel niet de klauw zelf, maar de mechanisme die het helpt in alle directies. In het geval dat je NIET weet hoe het werkt, leg ik het uit. Er is dus een stok en die is aan alle zijden aangesloten bij nog een stok, en op de middenste stok zit de klauw. Misschien is dat niet zo goed te begrijpen, dus hier is een schema:



Begrepen? Goed. Nou, dit ga ik dus de arm noemen

Want het is makkelijker dan pneumatische mechanische grijpermechanisme. Het arm is dus hetzelfde maar de ding die aan de arm VASTZIT niet. Dat is dus een inktvlesje. Dus de informatie van de brein

```
(00000111111111100000000111000000001110000011101100001101110000110011  
00001100110001110000000000001110001101100000011011000011101111111101  
11000001110000000011100000000111111111100000)
```

Wordt naar de arm gebracht en de arm gaat het weer in die vierkant veranderen.

```
(00000111111111100000  
00011100000000111000  
00111011000011011100  
00110011000011001100  
01110000000000001110  
00110110000001101100  
00111011111111011100  
00011100000000111000  
00000111111111100000)
```

Er zit een stukje foam eronder die verdeeld in vierkantjes is en hij moet de inkt opslurpen. Dus de arm gaat in een simpele algoritme, zoals je hier kunt zien:



*En als het een plek is waar een één moet zijn, geeft de inktvlesje een beetje inkt (genoeg om de hele bodem te bedekken, maar niet zoveel dat het alle inkt opmaakt) en als er een nul is... gebeurt er niks. En dan heb je ... een pixelprint op foam! En dat brengt ons naar de derde (en laatste) stap:*

## ***LEVEL DRIE: DE PRINT***

*Eigenlijk is dit de makkelijkste stap om te doen maar bijna de moeilijkste stap om uit te leggen.*

*Dus wij hebben nu een pixel print op foam, maar hoe brengen we die pixel print naar de papier? Nou, daar hebben we iets extra's voor nodig. Of tewel: de draaibalk! (de officiële naam is het niet, DAT ken ik niet.) het is eigenlijk een van die dingen aan het einde van een dril (hier is een plaatje).*

*Maar het is niet scherp aan het einde en het is helemaal bedekt met die draai-dingen. En de pixel print wordt dus in plaats gehouden met een metalen vierkantje. En aan het zijde staat een kleine plekje dat precies bij de draaibalk past. Dus als je de draaibalk erin doet en hem laat draaien gaat de foam pixel print omlaag en omhoog! Dan doe je gewoon een stukje papier in een gaatje en dan krijg jij...*



## ***LEVEL VIER: DE OUTPUT***

*In computertaal heet de output wat de computer jou aan het einde geeft. En hier, bij de pixelator, is dat...*

***EEN PIXLY  
TEKENING!!!!!!!!!!***

*Dit is het einde van de uitleg.*