

Inhoud

1	This is IT!	1
	Inleiding	2
	Hogepriesters en elite	3
	Divergeren, niet convergeren	4
	Digitale emancipatie	5
	De IT'er herschept de wereld naar zijn evenbeeld	7
	Even en oneven	9
2	Hello World!	11
	Integrated Development Environment	12
	Experimenten in een virtuele pc	12
	Replit	13
	Hello World in C	14
	Hello World in Python	17
	Lagere en hogere programmeertalen	18
	Kwik, Kwek en Kwak	19
	De wereld kijkt mee...	23
	Wat hebben we overgeslagen?	25
3	Bloem, gist & water	27
	Inleiding	28
	Communiceren met de computer	28
	Modelleren	29
	Onthouden met variabelen	30
	Ordenen	31
	Berekenen	32
	Beslissen met IF-ELIF-ELSE	32
	Herhalingen	33
	Zero-based tellen	35
	Wat kun jij met deze kennis?	36
4	Ingrediënten & experimenten	37
	Inleiding	38
	Variabelen	38
	Rekenen, ook met tekst	39
	Typecast str(), int()	40
	Print met toeters en bellen	41
	IF-THEN-ELSE	43
	Bugs in Replit: TabError	45
	FOR en range()	45
	WHILE	46

Inhoud

	Functies en methoden	48
	Een bugje in Replit	49
	Experimenteren	50
5	Geheugen, appels & peren	51
	Inleiding	52
	Datatypes	52
	Waarde of verwijzing?	54
	Nog meer appels en peren	55
	Datatypes in essentie	56
6	Encryptie	59
	Inleiding	60
	Cijfers en letters	60
	Een reeks van letters	62
	Versleutelen zonder te ontsleutelen	68
7	Algoritmen in het dagelijks leven	71
	Inleiding	72
	Stukjes van het geheel	73
	Wat kun jij met deze kennis?	75
8	Kerstbomen opzetten	77
	Inleiding	78
	Werken met de shell	78
	Branches	80
	Menselijk gedrag modelleren	81
	Zoek het lekker uit, je hebt alle code	82
	Voor de liefhebber: recursie	83
9	Ordening & recycling	85
	Inleiding	86
	Waarschuwing vooraf: we beschouwen de wereld	86
	Functies hergebruiken	86
	Klassen beschrijven een stukje wereld	88
	OOP	90
	Dynamiek en ordening met lijsten	90
	Verzamelingen	92
	Wat kun jij met deze kennis?	93

10	Functies, klassen & lijsten	95
	Inleiding	96
	Lees- en printfuncties maken	96
	Een module importeren	98
	Parameters	99
	List	100
	Dictionary	103
	Klassen, subklassen en objecten	106
	Project	109
11	The Magic Box	111
	Inleiding	112
	Filters	115
	Filmbewerking	117
	Wat kun jij met deze kennis?	117
12	Fotofilters bouwen	119
	Inleiding	120
	Filtersmod.py	121
	Waarom?	126
	Video	126
13	Data	127
	Inleiding	128
	Bevroren tijd	128
	Uniform formaat	129
	Uniforme taal	130
	Vragen stellen	130
	Wat kun jij met deze kennis?	131
14	Hoe overleef je de Titanic?	133
	Van CSV naar DB	134
	SQL-uitvoer	136
	Stoomcursus SQL	137
	Gemiddelden en totalen	138
	SQL-foutmeldingen	139
15	Van bospad naar snelweg	141
	Inleiding	142
	Wat moet je met een website?	142
	Van bospad...	143
	...naar digitale snelweg	143
	De levenscyclus van een webverzoek	144
	Cookies	145

Inhoud

	Geavanceerder	146
	Security by obscurity	147
	Responsive design en mobile first	149
	Wat kun jij met deze kennis?	149
16	Flessenpost voor Ajax	151
	Meteen aan de slag	152
	Het framework Flask	153
	Het HTML-menu aanpassen	154
	Route en def()	155
	Een HTML-template invoegen	155
	Ajax	156
	Een webformulier verwerken	157
	De levenscyclus van een webverzoek	160
17	Computer says no	161
	Inleiding	162
	MoSCoW	162
	404	163
	Testen	163
	SDLC	164
	Purpose thinking over process thinking	165
	Regel zes	165
	Vertrouwen in faalbaarheid	166
18	Fouten & grenzen	167
	Inleiding	168
	Invoerfouten opvangen met try	169
	Honderd procent controle op invoer	170
	None, niets, leeg, False en 0	172
	ListIndex out of range	174
	KeyError	174
	404	175
	Epiloog: IT does matter	177
	Index	179

Voorwoord

Ik ben soms de tel een beetje kwijt. Sinds 1998 rolden titel na titel met mijn naam erop van de drukpers. Geen proza, maar boeken over Windows, Photoshop, HTML, ach noem maar op. Later ook veel heruitgaven, nieuwe edities, nog meer vertalingen, gebundelde cursussen, en zo loopt het aantal snel op. Dit is mijn honderdste titel.

Veel leuke boeken over leuke onderwerpen, zoals de voorlaatste titel *Online flexwerken en geld verdienen (short.jeex.eu/2)*, samen gemaakt met mijn jongste zoon, die bij het schrijven van mijn allereerste boek nog niet eens bedacht was.

Dit boek voelt als het laatste. Een soort legaat, de culminatie van jaren werk, experimenteren, research, lesgeven en leren. Dit boek sluit niets af, maar is wel het resultaat van waar ik de laatste jaren mijn tijd aan besteed heb en waar mijn passie ligt. Dit boek heeft bovendien een maatschappelijke noodzaak. Het voelt als een missie.

Binnen het hbo-onderwijs verzorg ik een minor die bij de derde editie al zeventig studenten heeft uit heel Nederland: Creatief Programmeren voor Niet-IT Studenten. Een minor die geschikt is voor alle hbo-studenten, behalve dus studenten Informatica, en die hetzelfde doel heeft als deze honderdste titel: de magie uit het fenomeen *programmeren* halen. Het belang onderstrepen van digitale emancipatie.

This is IT!

Programmeren is op dit moment wat lezen en schrijven was in de middeleeuwen: iets voor ingewijden, voor de elite. Dankzij het middel om op afstand en niet-tijdgebonden te kunnen communiceren en publiceren, werd de voor-sprong van die elite op het gewone volk steeds groter. Programmeren is een middel om te communiceren met het allerhoogste: de computer, de cloud. Programmeertaal is niet anders dan elke andere taal zoals Nederlands of Engels. En je leert het door het te doen.

Vandaar ook het *experimenteren* in dit boek. In alle even hoofdstukken kan de lezer zelf stukken code maken en aanpassen om zo te ervaren wat die code doet. Dat is niet moeilijk of eng, er kan niets misgaan, het is wel heel veel *fun*. Inderdaad, jij gaat programmeren terwijl je wellicht nog nooit een regel programmeercode hebt gezien.

Die code vormt de koppeling tussen jou als mens en alles wat er om jou heen gebeurt wat geautomatiseerd is. En dat is tegenwoordig zo'n beetje alles. Dit

Voorwoord

boek laat zien waarom de *Computer says no* (short.jeex.eu/26), en waarom we dat fenomeen in onze samenleving moeten overstijgen.

Dit boek is geen cursus programmeren

Dit boek is dus geen cursus programmeren. Althans, niet een cursus waarin je leert programmeren met een beroepsperspectief. Dit boek geeft je kennis en inzicht in de mechanismen achter het programmeren. Dit boek gaat ook in op het *waarom* van IT en van allerlei *best practices*, en minder op het *hoe*.

Daarvoor is het hier en daar wel nodig om allerlei details te leren over programmeren. Wil je zelf de dingen kunnen ervaren, dan zul je ook wat zaken moeten kennen en begrijpen die nogal specifiek zijn. Alles wat je leert in dit boek heeft een functie, om zo tot een beter begrip te komen. Je leert zeg maar lezen en schrijven, maar verwacht niet dat je meteen novellist bent.

Omdat ik in dit boek niet al te diep wil graven, gaan sommige onderwerpen wat kort door de bocht. Puristen hebben wellicht het nodige aan te merken op zaken die in dit boek aan de orde komen. Er wordt veel overgeslagen, veel details worden niet benoemd, en heel veel vragen worden niet gesteld of beantwoord. Dat hoort er nu eenmaal bij als je een complex fenomeen begrijpelijk wilt maken. Dit zijn geen excuses, maar het is een keuze.

Hoe werkt dit boek?

Hoofdstuk 1 van dit boek begint met de zin: *Ik beloof je dat dit boek niet technisch is*. En die belofte houd ik het hele boek door. Maar, toegegeven: dat is natuurlijk subjectief en arbitrair. Doel is echter dat dit boek ook als een boek leesbaar moet zijn, bijvoorbeeld onderweg in de trein. Ook zonder bureau en laptop voor je neus. De oneven hoofdstukken zijn dus puur leesvoer, de even hoofdstukken zijn experimenten, maar ook gewoon leesbaar.

Bij sommige experimenten heb je een pakketje met voorgebakken code nodig. Het downloaden daarvan is ook grotendeels geautomatiseerd. Geen gestoei met zipjes en bestandjes dus. Alles is echt hapklaar.

Al het programmeren gebeurt in een browser. Dat kan Firefox zijn, Edge, Safari of Chrome. Maakt niet uit. Je hebt dus geen speciale software nodig in de computer. Elke computer kan dit.

Bedenk bij de experimenten in dit boek: je staat er niet alleen voor. De wereld is vol mensen die je met liefde en plezier verder helpen. Dat kan dichterbij zijn dan je denkt. Zoals op het Discord-kanaal dat voor dit boek is opgetuigd. Mij kun je daar ook vinden. Veel plezier. Ik zie je graag op Discord:
short.jeex.eu/discord-idm.

This is IT!

1 Hoofdstuk

In dit hoofdstuk:

Hogepriesters en elite.

Divergeren, niet convergeren.

Digitale emancipatie.

De IT'er herschept de wereld naar zijn evenbeeld.

Inleiding

Ik beloof je dat dit boek niet technisch is. De woorden die ik gebruik, betekenen wat ze zeggen, ook als ze Engelstalig zijn.

Toen ik in 1983 als kersvers studentje naar de HTS in Groningen ging om Informatica te studeren, stond het vakgebied nog in de kinderschoenen. Eigenlijk wist de HTS zelf nauwelijks wat de inhoud van die studie moest zijn. Zeven verschillende vakken wiskunde stonden op het programma en ook nog een paar programmeertalen. Cobol, Fortran en Algol 68 waren programmeertalen die als theorievak werden gegeven, want praktijk was niet mogelijk. Alleen Pascal – toen een heel moderne programmeertaal – werd daadwerkelijk in de praktijk uitgevoerd.

Maar op de keper beschouwd leerden we niets over programmeren. Geen aandacht voor *problem solving strategies*, niets over algoritmen, niets over testing, en ik denk nu dat de docenten daar toen zelf ook niets over wisten, want het vakgebied was simpelweg te nieuw en te obscuur om al geschoolde en ervaren docenten te hebben. Als student deden we maar wat en ik heb dus heel veel geleerd. Als ik terugkijk, mijmerend langs het tuinpad van mijn vader, bestond de praktijk op school vooral uit het hacken van het centrale computersysteem om zo meer computerminuten toebedeeld te krijgen. Want die waren op rantsoen. Niemand wist toen waar het heen zou gaan met IT.

Enkele jaren later, op de Academie voor de Journalistiek, liet een enthousiaste docent zien dat er zoiets bestond als internet. Hij gaf overigens het vak Nieuwe media, wat toen nog ging over tv en Teletekst, want dat was de toekomst: *media on demand*. Er waren geen websites, HTTP en HTML bestonden nog niet, maar die docent had toegang tot LexisNexis: een enorme database waar je van alles in kon vinden. Wat of hoe, dat weet ik niet meer. De docent in kwestie was zo aangedaan en lyrisch over het feit *dat* het kon, dat hij nauwelijks een voorstelling had of kon geven van *wat* of *hoe* het kon. Die herinnering is wezenlijk. Dat fenomeen overigens ook: een verwondering die zo groot is, dat je hersenen niet in staat zijn om er praktisch over na te denken; een werkelijkheid die niet te bevatten is, omdat je fantasie zo ver niet reikt.

Hogepriesters en elite

Eind 1998, toen mijn eerste boekvertalingen over Windows 98 en Office en dergelijke in de winkel lagen, struikelde ik in de boekwinkel (Gianotten in Tilburg) over de benen van een oudere dame die op de grond zat met een stapel computerboeken om zich heen. Ze was aan het kiezen tussen enkele titels over, jawel, Windows 98. Want zij vond toen al dat ze als senior actief mee moest gaan met de tijd. “Computer en internet zijn geen regenbuitje dat overwaait” was haar terechte voorspelling. En welk boek ze dan moest kopen, was haar vraag. Mijn antwoord: Visuele Leermethode Windows 98, want dat had ik vertaald en ik wist dat het een goed boek was. Royalty’s verdiende ik er niet mee.

Enfin, de strekking van het gesprek ligt voor de hand en ook welk boek ze meenam. Maar deze dame, inmiddels half overeind gekomen, nu knielend aan mijn voeten en naar me opkijkend, deelde mij mede, dat ik als IT’er in haar ogen en in haar woorden “de macht van de hogepriesters” bezat. Ik had als *uitverkorene* toegang tot de kennis die schuilgaat achter de werkelijkheid die de wereld gaat beheersen. En dat maakt mij deel van een elite die meer directe invloed heeft op onze directe leefwereld, dan alle politici en machthebbers bij elkaar.

De hogepriesters, zoals de dame ze noemde, de kerk, hield lang geleden het klootjesvolk graag dom. Kennis hoorde bij uitverkorenen en vooral niet bij iedereen. Alleen de kerk bepaalde welke kennis gedeeld werd en welke vooral niet. Dankzij die macht over het geschreven woord, bleef de aarde beduidend langer plat dan nodig was.

Terug naar de IT. Programmeurs zijn niet anders dan automonteurs, aannemers of ambtenaren. Ze zeggen graag dat iets niet kan, of dat iets heel duur en ingewikkeld is, om hun eigen luiheid of onvermogen te verbergen, om zichzelf belangrijk te maken, of gewoon om een dikkere rekening te kunnen sturen. Het feit dat IT’ers beduidend meer kennis van IT hebben dan de doorsnee burger, terwijl de burger niet meer zonder kan, geeft hen macht. In de wereld van IT is dat echter een macht die verder gaat dan een gepeperde rekening voor het inpluggen van een stekker of het veranderen van een stukje programmeercode.

Divergeren, niet convergeren

Nog even terug naar de late jaren 80 van de vorige eeuw. Sinds die tijd haalt de IT-sector mensen uit alle andere sectoren om programmeur, projectmanager en dergelijke in de IT te worden. Afgestudeerden Toegepaste Huishoud Wetenschappen, Vrijtijdskunde, Bedrijfskunde, noem het maar, werden met grote salarissen geworven en omgeschoold tot IT'er. Die convergentie richting IT was normaal dankzij de grote werkloosheid onder jongeren, en werd alleen maar erger toen de millenniumbug de wereld dreigde te verstoren.

Die *funneling* richting IT gebeurt heden ten dage nog steeds in de vorm van prima betaalde traineeships voor hbo- en wo-afgestudeerden. Dit is net zo goed een *brain drain* als de verhuizing van afgestudeerden vanuit de landelijke regio's richting Randstad. Professionals stromen richting IT en vergeten gaandeweg hun oorspronkelijke vakgebied, ook omdat ze daar in hun jonge leven nog niet echt hun sporen verdiend hebben.

Steeds meer sectoren werken met software die voorzien is van een scriptinglaag. Dat betekent dat je als gebruiker van die software de software kunt aanpassen aan jouw behoeften. En dat je meer uit de beschikbare data kunt halen dan de software standaard biedt. Dat biedt kansen. Leer je als expert in jouw vakgebied werken met deze scriptingtaal, wat op zich geen *rocket science* is, dan kun je meer betekenen voor jouw werkomgeving, en je maakt als persoon een sprong in je carrière. Het alternatief is dat je jouw lot op het hakblok van een willekeurige programmeur legt om die scripting voor jou te doen.

IT'ers kunnen niets

IT'ers kunnen op de keper beschouwd niets en weten ook niets van de wereld om zich heen, behalve hoe ze moeten programmeren. Dat bedoel ik minder lelijk dan het klinkt. Maar je kunt van een IT'er die een systeem voor eiwittenanalyse moet bouwen, niet verwachten dat deze eerst eiwittenexpert wordt. Als verkeersmanager in een grote stad, als medicus met complexe scanapparatuur, als bioloog in een eiwittenlab, moet je dus ook niet de resultaten van jouw werk laten beperken door wat een IT'er kan begrijpen van jouw werk.

Het aantal *hogere programmeertalen* – dat woord *hogere* licht ik verderop in dit boek toe, voor nu mag je het lezen als: *toegankelijke programmeertalen* – lijkt dagelijks te worden uitgebreid. Programmeren wordt steeds makkelijker in allerlei verschillende contexten, maar wordt ook steeds meer nodig. Denk aan Python om ArcGIS te programmeren, of aan SQL om data uit SPSS te halen. Beide zijn programmeertalen die iedereen kan leren. Misschien niet tot de volle diepte om zelf complete systemen te bouwen, maar diep genoeg om er

zelf wat mee te kunnen en om een goede gesprekspartner te zijn tussen vakbroeders en -zusters en programmeurs.

Dit boek is dus ook een oproep om IT-skills te divergeren en naar andere sectoren toe te brengen, niet andersom. Dit boek is een oproep aan studenten in mbo, hbo en wo: leer programmeren. Niet om programmeur te worden, maar om te leren communiceren met de computer en met softwareontwikkelaars.

Digitale emancipatie

Als software engineer, journalist en hbo-docent Informatica kan ik me prima verplaatsen in de onwetendheid van de doorsnee gebruiker van computers, websites en apps. Wat er achter die voorkant gebeurt, is voor de leek vaag, of eigenlijk compleet duister. Digitale apparaten en diensten zijn een soort *magic box* waarin iets gebeurt waar je geen zicht op hebt, maar waar je wel dagelijks, uurlijks en zelfs minutelijks (is dat nog Nederlands?) mee werkt. De hele wereld hangt aan het – al of niet draadloze – draadje, terwijl maar een kleine groep mensen weet wat er in die *magic box* gebeurt.

Dat is absurd en zelfs gevaarlijk. Directe problemen die daaruit voortvloeien zien we dagelijks in de vorm van fraude, oplichting en digitale afpersing. Bijna wekelijks krijg ik een berichtje dat mijn computer of mijn iPhone is gehackt, of dat mijn bank wil dat ik mijn identiteit even bevestig en of ik even bitcoins wil storten. Allemaal sneue en kinderlijk eenvoudige pogingen om mij geld afhandig te maken. Ik stink er niet in, want ik weet dat het op onzin berust. Maar het feit dat dit fenomeen toeneemt doet me vermoeden dat deze plakbandhackers er goed mee scoren bij onwetende en vooral onzekere mensen.

De fout, de oorzaak van die fraude, ligt niet bij de fraudeur, maar bij de onbenul die een apparaat koopt of een dienst gebruikt zonder enige kennis van zaken. Dat is niet aardig van me, maar we hebben voor wat betreft het werken met complexe apparatuur een dubbele moraal. Dit soort fraude is alleen maar mogelijk omdat het slachtoffer totaal niets weet over het apparaat waarmee hij of zij werkt. Het is alsof je een kind een fluxcapacitor in handen geeft en zegt: ga daar maar lekker mee spelen.



Nee, dit is niet overdreven. Ik ken geen enkel vakgebied, geen enkele branche, waar je als complete onbenul met een uitermate complex en gevaarlijk gereedschap zomaar aan het werk wordt gezet, en dan ook nog zonder enige uitleg of toezicht. Maar met IT-producten en diensten gebeurt dit dagelijks. We doen het zelf en kiezen er elke dag opnieuw voor. En we houden onszelf voor dat we niet zonder kunnen.

Moet ik de bijbehorende risico's nog noemen? Ach, doe eens gek. Online fraude, oplichting met niet-bestaande webshops, kinderen die in de fuik van een pedo lopen, phishingmails en vergeet vooral de door zoekalgoritmen aangetrokken nieuwsfuiken en het fenomeen nepnieuws niet. Onwetende mensen zijn er altijd geweest, maar de verstrekkende gevolgen van de fabeltjesfuik – zoals Lubach het noemt – is te wijten aan het kind dat met de fluxcapacitor mag spelen. Kijk nu eerst even in de spiegel, voordat je anderen uitlacht.

Emancipatie is het streven naar een volwaardige plaats in de samenleving vanuit een achtergestelde positie. Dat woord is uitermate relevant voor de positie van vrijwel iedereen die niet technisch inhoudelijk werkzaam is in het vakgebied van IT. Dat is een ongewenste positie waaruit je jezelf moet willen bevrijden. Nogmaals: kijk even in de spiegel. Dit gaat niet over mensen die zichzelf digibeet noemen, omdat ze onhandig zijn met knopjes. Dit gaat over het grootste deel van de samenleving, ook degenen die uitermate handig zijn in het bedienen van de knopjes en menen dat ze alles heel goed begrijpen.

Digitale emancipatie is een bittere noodzaak als we als mens volledig de beschikking over ons eigen lot willen houden. *Terugnemen* is wellicht beter van toepassing, want we hebben voor een heel groot deel de regie over onze dagelijkse werkelijkheid al overgedragen aan software waar we niets van begrijpen en geen enkele invloed op hebben.

Daarover gaat dit boek: software begrijpen. Daar begint ook in mijn ogen de digitale emancipatie: begrijpen en ervaren wat er in die *magic box* gebeurt, zodat je er een oordeel over kunt vellen en beslissingen over kunt nemen. Daarvoor hoeft je geen programmeur te worden.

De IT'er herschept de wereld naar zijn evenbeeld

Dagelijks zijn we overgeleverd aan systemen die taken moeten automatiseren. Digitale systemen. Dat woord *digitaal* is al enkele malen gevallen en is essentieel. Digitaal duidt oorspronkelijk op het tellen met vingers en gaat natuurlijk over het feit dat in een computer alles *telbaar* moet zijn. In een digitale wereld bestaan geen tussenwaarden en ook *misschien* kent een computer niet.

In de huidige tijd verwijst het woord digitaal vooral naar een veel enger woord: binair, tweewaardig. Die twee waarden zijn aan of uit. Vanuit die basis, 0 of 1, aan of uit, ja of nee, is de huidige nieuwe wereld opgebouwd. Twee voorbeelden schetsen de gevolgen van binairiteit en het volslagen gebrek aan analoge, menselijke mogelijkheden, ten gevolge van door ons bedachte systemen.

Kort na mijn studie journalistiek had ik kennis aan een student, van wie beide ouders overleden waren. Op zichzelf al naar genoeg. Op het formulier van de toen nog vrij nieuwe organisatie DUO kun je invullen of één van je ouders overleden was. Je mocht kiezen: was Pa overleden of Ma. Allebei kon niet op het formulier. De DUO-ambtenaar begreep na een telefoontje het probleem van mijn kennis: er was geen keuzevakje voor Pa *en* Ma. Maar de ambtenaar kon er niets aan veranderen: het formulier was immers het verlengstuk van het systeem. Pas negen maanden later kon mijn kennis een eerste studiebeurs ontvangen. Tot die tijd moest hij, aldus de brieven die het systeem volautomatisch uitbraakte, maar bij zijn ouders aankloppen.

Ik meen dat het de gemeente Utrecht was, waar een gezin genaturaliseerde Nederlanders – al decennia woonachtig in Nederland, kinderen hier gekregen en inmiddels op school – verhuisde van de ene straat naar de andere in dezelfde wijk. Natuurlijk was dat netjes aan de gemeente gemeld. Ruim een halfjaar later, door een bezoekje aan de huisarts, werd vastgesteld dat het gezin geen zorgverzekering meer had. Vreemd. Na het nodige uitzoeken bleek dat de gemeente het gezin terug naar het land van herkomst geremigreerd had, administratief dan. Een ambtenaar had het verkeerde vinkje in de computer aangeklikt. De gemeente kon deze administratieve fout niet herstellen, want het systeem... Dus bood ze als oplossing dat het gezin de hele immigratiecyclus opnieuw zou doorlopen. Dan kwam alles weer goed, voor de gemeente dan. *Computer says no...*

De computer, het systeem, bepaalt wat kan en wat niet kan. We maken onszelf ondergeschikt aan onze eigen automatisering. We hebben dat fenomeen ook zo in onze taal opgenomen: wij *bedienen de computer*. Zo zeggen we dat en zo zien we dat blijkbaar. Terwijl het toch echt omgekeerd moet zijn: de computer bedient ons. Het zijn maar woorden, maar ze zijn letterlijk gaan betekenen hoe

we het uitspreken. We hebben onszelf onderworpen aan systemen die we zelf bedenken en ontwikkelen.

Ik schrijf hierboven het woord *we* als het gaat over bedenken en ontwikkelen, maar wie zijn *we*? Precies: de uitverkorenen, de hogepriesters, de ingewijden. Zijn we als samenleving al te ver afgedaald in deze afgrond van esoterische digitalisering? Bepaalt dat kleine groepje IT'ers, waarvan menigeen heel wat vinkjes kan zetten op een vragenlijst over autisme, hoe onze wereld er uitziet? De IT'ers van vandaag herscheppen de wereld naar hun evenbeeld. Dat gaat verder dan wat administratieve systemen, digitale formulieren en websites met keuzevakjes.

Vrijwel onmerkbaar veranderen allerlei digitale processen in quasi intelligente volautomatische processen. Dat noemen we dan *artificial intelligence* of kunstmatige intelligentie, wat een containerbegrip is voor allerlei soorten van autonomie binnen software. Natuurlijk kan een digitaal proces heel goed gebouwd worden om te interpoleren. Heel veel menselijke processen zijn prima over te laten aan digitale uitvoerders. Wiskundigen kunnen berekenen met welke mate van nauwkeurigheid dat kan, wat de foutmarge is en hoeveel slechter mensen zo'n proces zouden uitvoeren. Zolang het bij interpoleren blijft.

Denk daarbij niet alleen aan processen in industrie, zoals lassen in een autofabriek of *order picking* in een pakhuis, maar denk ook aan het vervangen van een rechter in kleinere rechtszaken. Ook dat kan digitaal en misschien zelfs beter dan het nu gaat. Voor je verder leest zou ik je willen uitnodigen om met aandacht deze video met Hannah Fry – een Britse wiskundige – te bekijken. De video is zeer vermakelijk, maar ook uitermate informatief en relevant: short.jeex.eu/37.

Dat herscheppen van de wereld gebeurt op twee niveaus: met kunstmatige intelligentie beslissingen nemen, en in de vorm van robotica en hardware. Zowel *denken* als *handelen* krijgen dus nieuwe verschijningsvormen, die volledig uit de kokers van IT'ers en werktuigbouwkundigen komen. Dat zijn – kort door de bocht, ik wil niemand beledigen en ook zeker niet generaliseren – nerds die nabouwen wat ze 's morgens in de spiegel als een volslagen normaal wereldbeeld beschouwen.

De programmeur schept binnen de context van een opdracht dus een stukje wereld naar zijn of haar evenbeeld. Dat evenbeeld is al te vaak een spreekwoordelijke bezemkastnerd wiens volledige leefwereld van 8:00 tot 17:00 bestaat uit een zwart scherm met witte letters programmeercode en van 17:00 tot 8:00 uit World of Warcraft, om maar een game te noemen, met tussendoor wellicht een uurtje tijd voor slaap, pizza en Red Bull. Dit beeld is natuurlijk aangezet, maar helaas minder dan je zou wensen.

Dit is geen pessimisme, maar wel een realiteit die om aandacht vraagt. Dit is ook niet het moment om gemakzuchtig naar Den Haag of Brussel te wijzen en te zeggen dat *ze* er wat aan moeten doen. Schuif de verantwoordelijkheid niet af, want hier ligt voor jou een taak, beste lezer. De taak om te ontsnappen aan jouw digitale onbenulligheid. Jij hebt de verantwoordelijkheid om te gaan snappen wat er in die *magic box* gebeurt. Voor jezelf, maar ook voor jou als deel van het collectief dat we samenleving noemen.

Hier begint het! *This is it!* Ervaren wat zo complex lijkt, de magie ontrafelen en ontdekken dat software, programmeren, in wezen iets heel eenvoudigs en zelfs menselijks is.

Even en oneven

In het volgende hoofdstuk ga je bijzonder eenvoudige experimentjes doen met code. Je laat programmeertalen *Hello World* zeggen. Alle even hoofdstukken in dit boek bevatten experimenten die je kunt doen. De oneven hoofdstukken zijn minder praktisch en meer beschouwelijk van aard. Op elk beschouwelijk hoofdstuk volgt op die manier een hoofdstuk dat de besproken onderwerpen in praktijk brengt. Dit hoofdstuk was mijn *Hello World*, in het volgende hoofdstuk kun jij dat doen.

Ik probeer me aan mijn belofte te houden: dit boek is niet technisch. Het ligt niet aan jou, als je af en toe het gevoel hebt als in deze video: short.jeex.eu/biggestfear. Geef mij maar de schuld.