



Stichting NIOC

Stichting NIOC en de NIOC kennisbank

Stichting NIOC (www.nioc.nl) stelt zich conform zijn statuten tot doel: het realiseren van congressen over informatica onderwijs en voorts al hetgeen met een en ander rechtstreeks of zijdelings verband houdt of daartoe bevorderlijk kan zijn, alles in de ruimste zin des woords.

De stichting NIOC neemt de archivering van de resultaten van de congressen voor zijn rekening. De website www.nioc.nl ontsluit onder "Eerdere congressen" de gearchiveerde websites van eerdere congressen. De vele afzonderlijke congresbijdragen zijn opgenomen in een kennisbank die via dezelfde website onder "NIOC kennisbank" ontsloten wordt.

Op dit moment bevat de NIOC kennisbank alle bijdragen, incl. die van het laatste congres (NIOC2025, gehouden op donderdag 27 maart 2025 jl. en georganiseerd door Hogeschool Windesheim). Bij elkaar zo'n 1500 bijdragen!

We roepen je op, na het lezen van het document dat door jou is gedownload, de auteur(s) feedback te geven. Dit kan door je te registreren als gebruiker van de NIOC kennisbank. Na registratie krijg je bericht hoe in te loggen op de NIOC kennisbank.

Het eerstvolgende NIOC vindt plaats in 2027 en wordt dan georganiseerd door HAN University of Applied Sciences. Zodra daarover meer informatie beschikbaar is, is deze hier te vinden.

Wil je op de hoogte blijven van de ontwikkeling rond Stichting NIOC en de NIOC kennisbank, schrijf je dan in op de nieuwsbrief via

www.nioc.nl/nioc-kennisbank/aanmelden-nieuwsbrief

Reacties over de NIOC kennisbank en de inhoud daarvan kun je richten aan de beheerder:

R. Smedinga kennisbank@nioc.nl.

Vermeld bij reacties jouw naam en telefoonnummer voor nader contact.

Een integrerende benadering van de informatica in het voortgezet onderwijs en in het hoger onderwijs

Raymond Boute
Katholieke Universiteit Nijmegen
Toernooiveld 1
6525 ED Nijmegen

Samenvatting

Onderwijs in de informatica heeft voordeel bij een betere samenhang met de andere exacte vakken. In het VWO is dit mogelijk door integratie in de vakken wiskunde (wiskundige functies en functionele programma's) en natuurkunde; in het WO door voor de informatica de systeemtheorie als uitgangspunt te nemen, en omwille van de methodiek zowel aan natuurkunde als aan wiskunde een belangrijke plaats toe te kennen.

1 Inleiding

Het invoeren van de computer en de snelle ontwikkelingen van de Informatica hebben niet alleen een krachtig hulpmiddel ter beschikking gesteld van allerlei vakgebieden, doch hebben eveneens de denkwereld in belangrijke mate verrijkt. Het is echter misleidend de Informatica voor te stellen als een geheel nieuwe wetenschap die los staat van de andere exacte wetenschappen waaruit zij ontstaan is. De thans bestaande scheiding is een gevolg van historische en politieke omstandigheden, maar is epistemologisch niet gegrond en dient zelfs als schadelijk gezien te worden.

Deze laatste opmerking is gebaseerd op de waarneming dat in vele informatica-opleidingen de meer "klassieke" exacte wetenschappen uit het curriculum geweerd worden (de natuurkunde ontbreekt vrijwel overal, terwijl in sommige gevallen zelfs het wiskunde-aandeel ontoereikend is), met als gevolg dat de meeste studenten en afgestudeerden uit de informatica duidelijk merkbare moeite hebben met het zich eigen maken van denkwijzen en redeneerstijlen die bij andere exacte vakken zoals de natuurkunde gemeengoed zijn. Dit ondermijnt de bekwaamheid tot het opbouwen van wiskundige modellen, wat de eerste en meest essentiële stap is bij het aanpakken van een probleem uit de realiteit. Bovendien

betekent het zich uitsluitend beperken tot algoritmische processen een verarming van de denkwereld.

Eén van de oorzaken van dit verschijnsel is dat enerzijds informatici die deze meer klassieke exacte vakken wel vroeger in hun eigen opleiding hebben gevolgd de daarmee gepaard gaande verworvenheden als zodanig vanzelfsprekend zijn gaan beschouwen dat zij het doelbewust aanleren ervan niet als noodzakelijk zien, terwijl anderzijds informatici die deze vakken nooit hebben gevolgd vaak geen flauw benul hebben van waarover het gaat. De ondergeschikte rol die bijvoorbeeld aan de wiskundige analyse in vele informatica-opleidingen wordt toebedeeld is hiervan een typisch voorbeeld. Niet alleen is de wiskundige analyse van belang voor ieder informaticaprobleem dat men wil aanpakken op een professionele manier (met name: ook aandacht bestedend aan het kwantificeren en wiskundig modelleren van prestaties en betrouwbaarheid), doch is eveneens van fundamenteel belang als een epistemologisch voorbeeld van de manier waarop een brede en diepe kennisstructuur wordt opgebouwd, en van het modelleren van continue (en a fortiori van niet-algoritmische) processen.

Samenvattend zou men kunnen zeggen dat er, inhoudelijk gezien, even weinig reden was voor de Informatica om zich van de andere exacte vakken af te splitsen dan er, bijvoorbeeld, voor de quantummechanica reden was om zich van de natuurkunde af te splitsen ten einde zich te profileren als een nieuw vakgebied. Nu een dergelijke formele scheiding voor Informatica toch tot stand is gekomen, is de beste oplossing ernaar te streven dat de inhoudelijk nadelige gevolgen van die scheiding zoveel mogelijk beperkt blijven. Dit vereist een integrerende benadering, waarnaar de eerste stap gezet dient te worden in het informatica-onderwijs zelf.

2 De Informatica in het VWO

2.1 Uitgangspunten

Het VWO is bedoeld als voorbereiding op het wetenschappelijk onderwijs. De rol van de informaticavakken moet dus qua opvatting dezelfde zijn als die van de andere exacte vakken (wiskunde, natuurkunde, scheikunde, biologie), met name het geven van een eerste inzicht in de basisprincipes en de denkwijzen, alsook in het onderling verband (met betrekking tot inhoud en methodiek) tussen deze vakken. Zeker niet op hun plaats in het VWO zijn de volgende opties:

- Informatica bedoeld als inleidende programmeercursus en/of als kennismaking met een of andere programmeertaal. Immers, programmeren is een vak,

en net zomin als de natuurkunde in het VWO bedoeld is om tot technicus in de mechanica of de elektrotechniek op te leiden dient de informatica in het VWO vakbekwaamheden na te streven. Overigens maakt de beschikbare omvang voor dit vak het niet mogelijk boven het amateuristisch niveau uit te stijgen. Op dit laatste niveau wordt het vak immers door de "computerfreaks" onder de leerlingen voldoende buiten het onderwijs om beoefend.

- Informatica als gebruikerscursus voor een of ander programmapakket (zoals tekstverwerking of spreadsheets). Een passende vergelijking is hier dat natuurkunde in het VWO ook geen voorbereiding is op het rij-examen. Dit leert men net zo goed (zoniet beter) al doende, buiten de school.

Daarentegen vergt de rol van het vak informatica in het VWO wel aandacht voor:

a. Het verband met de wiskunde als basis, in het bijzonder de theorie van de berekenbaarheid, alsook de algoritmiek als middel om berekenbare functies met behulp van rekenregels te realiseren.

b. Inleidende algoritmiek (inclusief het afleiden van eenvoudige algoritmen uit eenvoudige specificaties) alsook enig besef omtrent concrete complexiteit (rekentijd, geheugenvereisten).

c. Het ondersteunen van de wiskundevakken door routine-rekenwerk eigenhandig te automatiseren op een manier die qua notatie en redeneerstijl nauw aansluit bij die van de wiskunde zelf.

d. Het ondersteunen van de natuurkundevakken door middel van eigenhandig ontwikkelde symbolische hulpmiddelen of numerieke simulatoren.

Daarbij dient men steeds vast te houden aan de discipline eerst een algebraïsche of analytische studie van het probleem te maken alvorens een geprogrammeerde oplossing te overwegen. Het denkwerk gaat vóór (of zelfs in plaats van) het programmeren, en potlood en papier gaan vóór de terminal. Deze prioriteitenkeuze moet een automatische reflex worden.

Bovendien dienen de informaticavakken in het VWO een soort tegengif te vormen tegen het onjuiste beeld van de informatica, de onsystematische aanpak en de verkeerde gewoonten die de "hackers" en "computerfreaks" zichzelf soms op jeugdige leeftijd hebben eigen gemaakt. Ook voor de leraren is het meer motiverend nieuwe principes bij te brengen dan zaken te behandelen die bij een deel van de leerlingen reeds bekend zijn.

2.2 Realisering

In dit stadium is het niet zinvol een volledig uitgewerkt informaticaprogramma voor het VWO voor te stellen. Wij beperken ons daarom tot de grote lijnen.

De lessen aangaande verzamelingen, relaties en functies uit het wiskunde-onderwijs dienen als vertrekbasis, en worden aangevuld met een verdieping van het functiebegrip. Om redenen waarop wij hier niet verder ingaan is het verkieslijk het begrip *functie* niet afhankelijk te maken van de begrippen Cartesisch product en relatie, zoals in de naïeve verzamelingenleer en sommige axiomatische formuleringen van de verzamelingenleer (zie Mendelson 1987 en Suppes 1972) gebruikelijk is, doch het in te voeren als een zelfstandig begrip, zoals bij de predicaten logica (zie Mendelson 1987). Het Cartesisch product en de relatie verschijnen dan als bijzondere gevallen. De hoger vermelde verdieping omvat de volgende twee punten:

- Het *typebegrip* met betrekking tot functies, waarbij $A \rightarrow B$ de verzameling (het "type") voorstelt van alle functies van een verzameling A naar een verzameling B . Na uitwerking en inoefening van dit eenvoudig begrip (met specialisering tot relaties, Cartesische producten en rijen) wordt benadrukt dat A en B in principe willekeurige verzamelingen zijn, dus eventueel ook functietypes. Dit leidt meteen tot het volgende punt.

- Het begrip *hogere-orde functie* als bijzonder geval van het functiebegrip (of als schijnbare veralgemening t.o.v. de tot op dat ogenblik behandelde voorbeelden) waarbij in het functietype $A \rightarrow B$ de verzamelingen van A en/of B zelf functietypes zijn.

De begrippen die verder de basis zullen vormen voor functionele programma's zijn op deze manier in het wiskunde-onderwijs geïntegreerd. Voorbeelden van "computer-uitvoerbare" functiedefinities worden regelmatig voorgesteld en gebruikt, met de belofte in een later stadium precies te omschrijven voor welke functies dit mogelijk is. Als functionele programmeertaal wordt Miranda voorgesteld (zie Turner 1985 en 1986).

Om meerdere redenen is het wenselijk de hoger beschreven verdieping van het functiebegrip in het onderwijs aan te bieden als een manier om wiskundige functies te gebruiken als *uitdrukkingsmiddel bij het formuleren van en het redeneren over problemen*. Concrete voorbeelden, in systematische volgorde opgebouwd, vormen aldus de motivering en de illustratie voor de ingevoerde begrippen.

De stap naar de algoritmiek ligt nu voor de hand via het functioneel programmeren. Vooreerst wordt aangetoond dat bepaalde uitdrukkingen in functiedefinities, naast hun gebruikelijke wiskundige betekenis, ook nog geïnterpreteerd kunnen worden als *rekenregels*. De deelverzameling van de *berekenbare functies* wordt dan gedefinieerd op de gebruikelijke manier, met name door het invoeren van drie basisfuncties (de nulfunctie, de ophoogfunctie en de projectiefunctie) en drie

samenstellingsregels (zuivere samenstelling, eenvoudige recursie en minimalisering), die alle als rekenregels opgevat kunnen worden. Een op deze manier opgebouwde functiedefinitie kan dan gezien worden als een *algoritme* op voorwaarde dat de uitvoering van de overeenkomstige rekenregels slechts een eindig aantal stappen vergt (terminatie).

Verdere uitwerkingen en verfijningen, zoals de definitie van functies over lijsten, het gebruik van functiedefinities voor de specificatie van en als hulpfunctie bij de realisering van algoritmen, liggen nu voor de hand. In dit stadium heeft de leerling een vrij volledig beeld van het wezen en de mogelijkheden van functioneel programmeren.

Het hierboven beschreven programma is, zelfs als men rekening houdt met de omvang van de informaticavakken en het niveau van het VWO, onvoldoende om de ruimte die in het VWO gemiddeld voor informatica beschikbaar is geheel op te vullen. Overigens is het duidelijk dat deze bespreking alleen de onderdelen a en b omvat uit de paragraaf "uitgangspunten". De invulling van de punten c en d is specifiek voor de gedoceerde wiskunde- en natuurkunde-vakken, en kan slechts door onderling overleg tussen de docenten tot stand komen.

In het bijzonder bieden vakken zoals natuurkunde en scheikunde veel aanknopingspunten om ook de informatica daarin te integreren, met name

- Op het gebied van *principes* en *theorie*, in de vorm van wiskundige modellen en daaruit afgeleide hulpprogramma's voor symbolisch of numeriek uitwerken van illustratieve problemen alsook het simuleren van verschijnselen, het bevattelijk voorstellen van resultaten e.d..

- Op het gebied van de *praktijk*, het *experimenteren* en het *meten*, in de vorm van hulpprogramma's voor het ontvangen van meetgegevens uit sensoren, het verwerken van gegevens en het aansturen van instrumenten. In de handel zijn ontelbare varianten beschikbaar van interface-kaarten die met weinig moeite aan de laboratoriumapparatuur gekoppeld kunnen worden.

3 De Informatica in het WO

3.1 Uitgangspunten

De rol van de wiskunde in de informatica-opleiding kan nauwelijks overschat worden. Ook de vakken omtrent de theorie en de grondslagen van de informatica dienen de kern te vormen van iedere informatica-opleiding. Daarnaast gelden echter ook de volgende overwegingen.

- a. De Informatica is in eerste instantie een op de toepassingen gerichte wetenschap. Dit geldt zelfs voort

de kerninformatica: de theorie van formele talen, de semantiek, de complexiteitstheorie enz. zijn alle gemotiveerd door de vertalerbouw en door de toepassingen van de algoritmiek in specifieke probleemgebieden. Daarom biedt de "zuivere" informatica onvoldoende basis en inhoud om de informatici op wetenschappelijk of industrieel vlak evenwaardig te maken aan niet-informatici die een degelijke opleiding hebben genoten in een ander vakgebied en de nodige informatica-kennis door zelfstudie hebben verworven. Een serieuze aanvulling door vakken met een ruime en diepgaande traditie in de wiskundige modelvorming (het weergeven van en redeneren over een aspect van de realiteit met behulp van de wiskunde) is daarbij echter onontbeerlijk. In de Informatica is deze kunst nog weinig ontwikkeld, terwijl de natuurkunde daarentegen het voorbeeld bij uitstek vormt, zowel voor wat de successen als voor wat de tegenslagen betreft. Om deze reden kan zelfs een vierjarig informatica-curriculum winnen aan blijvende waarde door enkele minder uitgekristalliseerde of minder inzicht-gevende onderdelen uit het hoofdvak zelf enige plaats te laten inruimen voor de bedoelde vakken.

Terloops weze hier vermeld dat men op langere termijn uiteraard dient te streven naar een vijfjarig curriculum. De overheid kan immers niet blijvend de vergissing van een vierjarige eerste fase in stand houden door de aanbevelingen van wetenschappelijke adviesraden en rectores (zie Hooge 1987) steeds opnieuw naast zich neer te leggen en de situatie in andere EEG landen te negeren. In 1992 gaan de grenzen officieel open voor buitenlandse diploma's, en met de huidige eerste fase heeft de Nederlandse doctorandus qua niveau een zeer ongunstige positie in vergelijking met universitair afgestudeerden uit de landen waar een vijfjarige studieduur met een inhoudelijk zwaarder programma geldt.

b. Niet alleen het onderwijs doch ook de kwaliteit van het onderzoek zijn gebaat bij een groter aandeel van de natuurkunde en van de abstracties die in de toegepaste wetenschappen gebruikt worden bij de wiskundige studie van signalen en systemen. Immers, sommige theorieën die in de informaticalliteratuur beschreven staan komen neer op het moeizaam herontdekken van wetenschappelijke methodieken die in de zojuist genoemde wetenschapsgebieden als gemeengoed gelden en aldaar veel eleganter en helderder geformuleerd zijn. Overdreven aandacht voor metaproblemen verhindert vaak het doordringen in de diepte: dit laatste vereist het ontwikkelen of overnemen van specifieke theorieën voor de beschouwde probleem- of toepassingsgebieden. De vergelijking met de andere exacte wetenschappen, alsook met onderdelen van de informatica waar dergelijke specifieke theorieën met succes zijn uitgewerkt (formele

talen, complexiteit) tonen aan dat de discrete wiskunde en de algebra voor de verdere ontwikkeling van de informatica de meest vruchtbare voedingsbodem bieden. Uiteraard zullen daarnaast de predicaatenlogica (zie Mendelson 1987), de verzamelingenleer (zie Suppes 1972) en de lambda calculus (zie Barendregt 1984) steeds belangrijk blijven voor de informatica. Met het oog op verdere ontwikkelingen dient men echter ook het volgende te bedenken:

- De problematiek van de grondslagen van de wiskunde uit de dertiger jaren (Gödel, Turing) die geleid heeft tot de eerste berekeningsmodellen voor de Informatica speelt thans een ondergeschikte rol: complexiteit staat meer centraal dan berekenbaarheid;

- De syntactische stijl van axiomatisering die ook nu nog door sommige informatici bij de theorie-ontwikkeling wordt aangekleefd dient plaats te maken voor een meer algebraïsch gerichte axiomatiseringsstijl zoals gebruikelijk is in de wiskunde bij het bestuderen van algebraïsche structuren (groepen, ringen, lichamen enz.). Deze biedt immers hetzelfde abstraherend vermogen met een aanzienlijk minder hinderlijke nomenclatuur (zie Boute 1988).

Het recente werk van Bird en Meertens (zie Meertens 1986 en 1989) illustreert duidelijk de voordelen van de algebraïsche stijl bij het transformationeel redeneren over specificatie en realisatie van discrete systemen en algoritmen. Ook het in voorbereiding zijnde boek van Dijkstra (zie Dijkstra e.a. 1990) toont aan hoe ver men kan komen met een sobere notatie en een eenvoudige bewijsstijl, gesteund op een algebraïsche vorm van de predicaatenlogica, bij het systematisch afleiden van en het redeneren over imperatieve programma's.

3.2 Realisering

In het WO kan de omvang van het informatica-onderwijs variëren van bijvakken tot een volledige studierichting. Het bespreken van curricula in hun geheel valt echter buiten het bestek van deze nota. Wij beperken ons daarom tot de integratie met de andere exacte wetenschappen waarvan de informatica als vakgebied technisch of epistemologisch afhankelijk is, of die tot de eruditie van iedere informaticus op academisch niveau moeten behoren.

Deze integratie betreft voornamelijk de basisopleiding (2,5 jaar), waarin drie componenten aanwezig zijn:

- a. Een zeer degelijke wiskundige component (algebra, analyse, stochastiek, logica, discrete wiskunde enz.) met als relatieve omvang ongeveer de helft van het totaal.

b. Een component gericht op de algoritmische benaderingswijze, met dezelfde uitgangspunten als eerder uiteengezet voor het VWO (het wiskundig functiebegrip, met als specialisering de berekenbare functies), thans echter op universitair niveau en aangevuld met andere formuleringswijzen (o.m. imperatieve) en andere abstractieniveaus (inclusief de lagere niveaus die verband houden met machinerealisering). Basisvakken zoals grondslagen van de informatica, berekenbaarheid, berekeningsmodellen, complexiteit, semantiek horen hier bij het verplichte deel van de opleiding.

c. Een component gericht op de niet-algoritmische benaderingswijzen in de exacte wetenschappen, bij uitstek vertegenwoordigd door de natuurkunde. Dit is bedoeld als tegengewicht tegen de operationele en discrete denkwereld van de algoritmiek, en leidt tot een vollediger beeld van de wiskundige modelvorming.

Onderdelen b en c dienen ongeveer dezelfde omvang te hebben. Er is aldus voldoende ruimte om de student gedurende de basisopleiding de voornaamste vakken uit de natuurkunde-propaedeutische te laten volgen.

Van belang is echter de overkoepelende rol van de systeemtheorie om de algoritmische en de niet-algoritmische denkwerelden (benaderingswijzen, modellen, ...) in een gemeenschappelijk kader te plaatsen. De klassieke systeemtheorie, met als basis: functies (in discrete en continue tijd) voor signalen, hogere-orde functies voor systemen, en een gedegen uitwerking van het begrip "toestand", vertegenwoordigt het algemeen geval van systemen die niet noodzakelijk algoritmisch zijn. Voor algoritmische systemen worden hieruit de nodige vereenvoudigingen (zoals de overgang van ware tijd naar partiële ordening) en specialiseringën afgeleid. In deze context zijn bijvoorbeeld datastromen een specialisering van signalen, en berekenbare functies (of Turing machines) een specialisering van systemen. Ook de overeenkomstige specialiseringën van het begrip toestand (onder meer bij imperatieve programma's) worden uitgediept. De benaderingswijze voorgesteld door Dijkstra (zie Dijkstra e.a. 1990) past uitstekend in dit kader.

Of deze overkoepelende rol van de systeemtheorie gerealiseerd wordt door middel van een afzonderlijk vak als onderdeel van component c, danwel via een "rode draad" doorheen alle vakken die deel uitmaken van de componenten b en c wordt in het midden gelaten. Die keuze hangt af van de mate waarin onderlinge afstemming tussen de docenten of herziening van reeds bestaande vakken in de curricula mogelijk is.

Wij benadrukken nogmaals dat wij ons in deze bespreking beperkt hebben tot die vakken in het curriculum die als basis moeten dienen en de informatica

als wetenschappelijke discipline kenmerken. Andere onderdelen van het curriculum, zoals

- kerninformatica (vertalerbouw, bedrijfssystemen)
 - databanken, informatiesystemen, administratieve toepassingen, bedrijfskunde,
 - op de technische toepassingen gerichte vakken,
- hoeven niet van meet af aan in dit integratieproces te worden betrokken, en kunnen dus in de meeste curricula in hun huidige vorm behouden blijven. Deze onderdelen plukken immers ook onrechtstreeks de vruchten van de integratie die in de basisopleiding gerealiseerd wordt.

4 Afsluiting

Om uitweidingen over varianten te voorkomen werden alleen het VWO en het WO besproken. Voor andere vormen van voortgezet onderwijs en hoger (beroeps)onderwijs kan de hier voorgestelde aanpak echter ook belangrijke voordelen bieden.

Gebruikte literatuur

- Barendegt, H.P. (1984) *The Lambda Calculus, it syntax and semantics*. Amsterdam: North-Holland
- Boute, R.T. (1988) On the shortcomings of the axiomatic approach as presently used in Computer Science. In: Proc. IEEE COMPEURO 88, System Design: concepts, methods and tools, 184-193
- Dijkstra, E.W. & C.S. Scholten (1990) *Predicate Calculus and Program Semantics*. New York: Springer Verlag, Inc.
- Hooge, F.N. (1987) *Onderwijsbeleid*. Rede uitgesproken bij de opening van het academisch jaar 1987-1988 van de Technische Universiteit Eindhoven op 7 september 1987
- Meertens, L. (1986) Algorithmics. In: J.W. de Bakker et al., eds., *Mathematics and Computer Science*. Amsterdam: North-Holland, 289-334
- Meertens, L. (1989) Constructing a calculus of programs. In: J.L.A. van de Snepscheut, ed., *Mathematics of program construction*, LNCS 375, Springer-Verlag, 66-90
- Mendelson, E. (1987) *Introduction to Mathematical Logic*, 3rd edition. Monterey: Wadsworth & Brooks
- Suppes, P. (1972) *Axiomatic set theory*. New York: Dover
- Turner, D.A. (1985) Miranda: a non-strict functional language with polymorphic types. In: J.P. Jouannaud, ed., *Functional programming languages and computer architecture*, Springer LNCS 201, Springer-Verlag, Berlijn, 1-16
- Turner, D.A. (1986) An overview of Miranda. *Sigplan Notices* 21, 12, 157-166