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 (NIOC2023, gehouden op donderdag 30 maart 2023 jl. en georganiseerd door NHL Stenden Hogeschool). Bij elkaar bijna 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 op donderdag 27 maart 2025 in Zwolle en wordt dan georganiseerd door Hogeschool Windesheim. Kijk op www.nioc2025.nl voor meer informatie.

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.

Het ontwerp van een universitair informatica-curriculum

A. Kaldewaij en C. Hemerik
Technische Universiteit Eindhoven
Postbus 513
5600 MB Eindhoven

Samenvatting

In dit artikel wordt een methode voor het ontwerpen van een universitair informatica-curriculum behandeld. De studie wordt verdeeld in een basisdeel en een bovenbouw. Voor het invullen van het basisdeel wordt stapsgewijze verfijning toegepast. Vanuit een vage specificatie wordt in een aantal stappen een concreet ontwerp verkregen. Elke stap wordt getoetst aan voorgaande resultaten. Het eindresultaat is een evenwichtig curriculum dat in 1989 van start is gegaan.

1 Inleiding

In 1982 ging de eerste-fase opleiding tot informatica-ingenieur aan de TUE van start. Het toen opgestelde studieprogramma kwam tot stand in een overleg tussen de faculteiten Bedrijfskunde, Electrotechniek en Wiskunde. De opleiding trok grote aantallen studenten en in de jaren die volgden werd de informaticastaf aanzienlijk uitgebreid, nam het aantal colleges toe en wijzigde de inhoud van bestaande colleges. Vijf jaar later bleek dit proces te hebben geleid tot verbrokkeling van en overlappingen in de opleiding en tot aansluitproblemen in de laatste fase van de studie. Een aantal tekortkomingen in het basisdeel van de studie (de eerste twee jaar) werden gesignaleerd, zoals

- Een onvoldoende samenhang tussen de vakken onderling.
- Onvoldoende kennis op het gebied van de theoretische informatica.
- Tekort aan bewijsvaardigheid en een tekort aan ervaring in het omgaan met formalismen.
- Onvoldoende vaardigheid in het implementeren van algoritmen.
- Het selecterend en oriënterend karakter van het eerste

jaar was onvoldoende.
- De studielast was niet homogeen verdeeld: het eerste jaar was te licht en het tweede jaar was te zwaar.
- Studenten schoten tekort in zowel mondelinge als schriftelijke uitdrukkingsvaardigheid.
- Voorkennis voor vakken in het derde en vierde studiejaar ontbrak.

Besloten werd dat een geheel nieuw informatica-curriculum moest worden opgezet. De auteurs van dit artikel waren toentertijd bestuurslid van de faculteit Wiskunde en Informatica en hebben in de ontwikkeling van dat curriculum een leidende rol gespeeld.
In het navolgende wordt uiteengezet hoe het nieuwe informatica-curriculum tot stand is gekomen. Een bijzonder aspect hierbij is dat de wijze waarop dit heeft plaatsgevonden analoog is aan de manier waarop systemen in de informatica worden ontworpen. De methode is vrij algemeen en de voor de TUE specifieke invulling van de diverse onderdelen is illustratief.

1.2 Werkwijze

Een mogelijke methode om tot de invulling van een curriculum te komen is de volgende:
- Formuleer de doelstellingen van de opleiding.
- Leid uit de doelstellingen eindtermen af waarin de kennis en kunde zijn beschreven.
- Bepaal uit de eindtermen de inhoud van het curriculum.

Deze weg is niet gekozen. Universitair onderwijs is door de interactie met het onderzoek voortdurend aan wijziging en vernieuwing onderhevig. Deze interactie manifesteert zich het sterkst in de specialisaties en veel minder in de voor alle studenten gemeenschappelijke basiskennis. Het is daarom zinvol de studie te verdelen in een basisdeel, gemeenschappelijk voor alle studenten, en een bovenbouw waarin specialisatie optreedt. Deze delen vergen elk hun eigen ontwerpmethode. In dit artikel wordt de uitwerking van een basisdeel besproken. De aanpak hiervan bestaat uit de volgende stadia:
1 Formulering van kenmerken van informatica, meer specifiek de informatica aan de T.U.E.
2 Afbakening van het basisdeel.
3 Verdeling van dit basisdeel in een aantal samenhangende deelgebieden.
4 Aangeven van onderwerpen per deelgebied welke relevant worden geacht voor het basisdeel.
5 Aangeven van de onderlinge samenhang der deelgebieden.
6 Toekennen van relatieve gewichten aan elk der deelgebieden.
7 Doen opstellen van gedetailleerde leerstofomschrijvin-

gen van de onderwerpen van de deelgebieden.
8 Implementatie van de leerstof: het afbeelden op trimesteruren en de keuze van onderwijsvormen.

Deze aanpak komt overeen met het ontwerpen van systemen. De eerste stappen zijn algemeen van aard, de middelste stappen leveren concretere randvoorwaarden en de laatste stappen leiden tot het product. Steeds kan na uitvoering van enige stappen het resultaat worden bediscussieerd, bijgesteld en vervolgens gefixeerd alvorens met volgende stappen wordt voortgegaan. Door deze stapsgewijze methode wordt verstrengeling van principes, onderwijskundige aspecten en vaktechnische argumenten voorkomen. Doordat de verschillende stadia worden besproken in een "brede discussie" wordt bovendien een stevig draagvlak voor het nieuwe curriculum verkregen. In de volgende paragrafen worden de bovengenoemde stadia uitgewerkt voor de specifieke situatie van de informatica-opleiding aan de T.U.E.. Deze uitwerking heeft twee jaar in beslag genomen.

2 Uitwerking

2.1 Formulering van kenmerken van informatica

In deze paragraaf geven we enkele uitspraken over informatica waarover binnen de vakgroep Informatica aan de T.U.E. consensus bestaat. Tevens geven we de consequenties voor een ingenieursopleiding aan dit instituut aan.

- Informatica is de wetenschap die zich bezighoudt met het beschrijven en ontwerpen van gegevensverwerkende systemen en met het ontwikkelen van theorie, methoden en technieken voor het construeren van dergelijke systemen.
- Genoemde systemen zijn artefacten, in tegenstelling tot bijvoorbeeld objecten die worden bestudeerd in de fysica. Dit betekent dat de nadruk niet ligt op het beschrijven en modelleren van het bestaande, maar veeleer op het ontwerpen van nieuwe systemen geleid door het streven naar correctheid, regelmatigheid en eenvoud.
- Informatica is bij uitstek een wiskundige discipline: alleen door geformaliseerde methoden en strikt wiskundig redeneren kan de complexiteit van deze artefacten worden beheerst.
- In de informatica tracht men, als in elke wetenschap, algemeen geldende uitspraken te doen die onafhankelijk zijn van specifieke toepassingen.
- Op een technische universiteit ligt de nadruk op de ontwerpaspecten van de informatica: het is een creatieve bezigheid waarbij inventiviteit een belangrij-

ke rol speelt.
- Informatica bevat voldoende aspecten om als volwaardig ingenieursvak te bestaan.

Bovenstaande uitspraken duiden op een attitudevorming die in de gehele opleiding een rol dient te spelen. Inzicht en vaardigheid in ontwerpen is belangrijker dan feitenkennis. Het gaat om een wetenschappelijke opleiding waarin het kweken van een kritische houding centraal staat.

2.2 Afbakening van het basisdeel

Het basisdeel van de opleiding dient aan de volgende voorwaarden te voldoen:
- Het omvat de kennis en kunde die tot de strikt noodzakelijke bagage van de informatica-ingenieur behoren.
- Het legt de grondslag voor elk der specialisatierichtingen van de opleiding en stelt de student in staat een bewuste keuze voor een specialisatierichting te maken.
- Het eerste jaar bevat voldoende stof om enerzijds de student in staat te stellen zich een oordeel te vormen over de informaticastudie en zijn geschiktheid daarvoor (het oriënterende aspect), en anderzijds de docenten in staat te stellen een verantwoorde selectie toe te passen (het selecterende aspect).

Belangrijkste uitgangspunt voor het vervolg is de eerste voorwaarde: de kennis en vaardigheden die tot de strikt noodzakelijke bagage van de informatica-ingenieur behoren. Nadat hiervan een invulling is gegeven kan aanvulling ten behoeve van de tweede voorwaarde plaatsvinden. Het derde punt komt aan de orde bij het afbeelden van de leerstof op trimesters.
Naast de beroepsgerichte vakken kent de TUE ook algemeen vormende vakken. Voor een goede appreciatie hiervan is een zekere rijpheid nodig, reden om deze als keuzevak in de specialisatiefase op te nemen.
Om het oriënterend aspect te versterken worden een P-colloquium en een D-colloquium ingesteld. Het P-colloquium biedt een beeld van het vervolg van de studie en van de toekomstmogelijkheden van een informatica-ingenieur. Het D-colloquium geeft voorlichting over de specialisatierichtingen.

De nominale studieduur is vastgelegd in de wet en bedraagt thans 4 jaar. Rekening houdend met een afstudeerperiode en een stageperiode is 3 1/3 jaar beschikbaar voor regulier onderwijs. Gezien de ervaring zal het specialisatiedeel daarvan 1 jaar nodig hebben. Bij verdere uitwerking wordt uitgegaan van een basisdeel

van 2 1/3 jaar (7 trimesters). Als laatste merken we op dat specialisatierichtingen vakken gemeenschappelijk hebben en dat de gemeenschappelijke kennis van de afgestudeerden zich niet beperkt tot het basisdeel. Ook zal binnen het basisdeel ruimte voor een keuzevak van een andere studierichting moeten zijn om al te eenzijdige vorming te voorkomen.

2.3 Verdeling in deelgebieden

Om een globale indeling van het basisdeel mogelijk te maken, waarbij relevant geachte onderwerpen een rol spelen en niet zozeer specifieke vakken gekoppeld aan aantallen trimesteruren, verdelen we het basisdeel in een aantal hoofdgebieden. Deze staan niet los van elkaar, integendeel: een goede onderlinge samenhang is essentieel.
Een keuze is gemaakt voor de volgende gebieden:
- theoretische informatica
- wiskunde
- programmeren
- computerarchitectuur en systeemprogrammatuur
- informatiesystemen

In het onderstaande wordt van elk der gebieden de rol aangegeven binnen het basisdeel. Let wel, het gaat hier niet om een opsomming van onderwerpen maar om het doel van het gebied.

Theoretische informatica
De theoretische informatica verschaft een algemeen referentiekader en levert een solide basis voor tal van constructies en methoden die in de andere gebieden aan bod komen. De theorie is unificerend en bevordert abstract denken. Ze is geen doel op zich maar steeds gekoppeld aan andere deelgebieden.

Wiskunde
Wiskunde levert inzicht in structuren en biedt methoden die geschikt zijn voor het oplossen van een grote klasse van problemen. Ook fungeert wiskunde als communicatiemiddel tussen diverse disciplines. Het wiskunde-onderwijs dient bij te dragen aan de bewijsvaardigheid en het omgaan met formalismen.

Programmeren
Het programmeeronderwijs is sterk attitudevormend. Het wordt van essentieel belang geacht voor iedere informaticus, zowel voor wat betreft de methode als voor de concrete uitvoering. De methode van programmeren kenmerkt de "Eindhovense School".
Computerarchitectuur en Systeemprogrammatuur
In dit vakgebied wordt inzicht verschaft in opbouw en

functioneren van rekenautomaten en de daarvoor specifieke programmatuur. Het illustreert de noodzaak van complexiteitsbeheersing en vormt een basis voor het ontwikkelen van andere systemen.

Informatiesystemen
Dit gebied houdt zich bezig met applicatiegerichte aspecten van gegevens-verwerkende systemen, in het bijzonder met het specificeren hiervan op conceptueel niveau. Het biedt een algemeen kader waarbinnen formeel over informatiesystemen wordt geredeneerd.

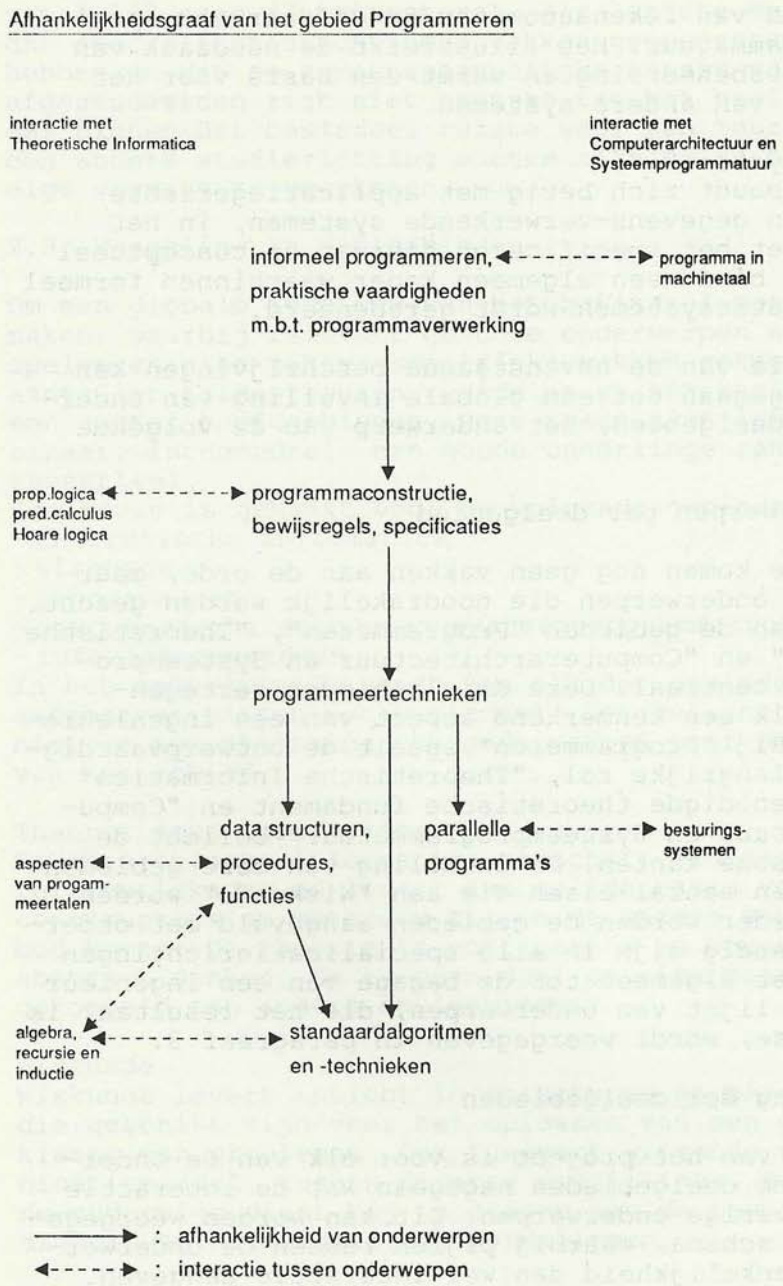
Na acceptatie van de bovenstaande beschrijvingen kan worden overgegaan tot een globale invulling van onderwerpen per deelgebied, het onderwerp van de volgende paragraaf.

2.4 Onderwerpen per deelgebied

In deze fase komen nog geen vakken aan de orde, maar betreft het onderwerpen die noodzakelijk worden geacht. Hierbij staan de gebieden "Programmeren", "Theoretische Informatica" en "Computerarchitectuur en Systeemprogrammatuur" centraal. Deze drie gebieden vertegenwoordigen elk een kenmerkend aspect van een ingenieursopleiding. Bij "Programmeren" speelt de ontwerpvaardigheid een belangrijke rol, "Theoretische Informatica" biedt het benodigde theoretische fundament en "Computerarchitectuur en Systeemprogrammatuur" belicht de meer praktische kanten. De invulling van deze gebieden leidt tot een aantal eisen die aan "Wiskunde" worden gesteld. Verder worden de gebieden aangevuld met onderwerpen die nodig zijn in alle specialisatierichtingen of die in het algemeen tot de bagage van een ingenieur behoren. De lijst van onderwerpen, die het resultaat is van deze fase, wordt weergegeven in paragraaf 3.

2.5 Samenhang der deelgebieden

In dit deel van het project is voor elk van de onderwerpen uit de deelgebieden nagegaan wat de interactie is met de overige onderwerpen. Dit kan worden weergegeven met een schema, waarbij pijlen tussen de onderwerpen een afhankelijkheid dan wel interactie aangeven. Een voorbeeld van een dergelijk schema is gegeven in Figuur 1. Deze schema's vormen een belangrijk hulpmiddel bij het structureren van de deelgebieden. Zij zijn ook het uitgangspunt voor de in fase 7 op te stellen gedetailleerde leerstofomschrijvingen (zie paragraaf 2.7).



Figuur 1
Samenhangende deelgebieden

2.6 Relatieve gewichten der deelgebieden

De omvang van de gebieden is afhankelijk van de veelheid van onderwerpen en van de gewenste diepgang. Zo zal Wiskunde door de veelheid van onderwerpen een aanzienlijk deel in beslag nemen, terwijl bij Programmeren en Theoretische Informatica de diepgang dit bewerkstelligt. Op grond van de inventarisatie van 2.4 wordt de volgende indicatieve verdeling voorgesteld:

Theoretische Informatica	20%
Wiskunde	30%
Programmeren	25%
Computerarchitectuur en Systeemprogrammatuur	20%
Informatiesystemen	5%

De uitwerking tot en met dit gedeelte is voorgelegd aan de vakgroep Informatica, het faculteitsbestuur en de faculteitsraad. De goedkeuring garandeert dat de vele tijd die gemoeid is met de verdere, meer technische, uitwerkingen niet vergeefs wordt geïnvesteerd. Bij die uitwerkingen kan bij meningsverschillen op het voorgaande worden teruggegrepen. Ook kan hobbyisme van een individuele docent eenvoudig worden bestreden.

2.7 Gedetailleerde leerstofomschrijvingen

Voor de uitvoering van dit onderdeel is per deelgebied een "voortrekker" aangewezen. Een voortrekker heeft goede contacten met de docenten op zijn gebied en voelt zich verantwoordelijk voor het onderwijsgebeuren. Uitgaande van de onderwerpen per deelgebied (paragraaf 2.4), de interactieschema's (paragraaf 2.5) en de relatieve gewichten (paragraaf 2.6) heeft zo'n voortrekker een indeling in vakken gemaakt en per vak een gedetailleerde leerstofomschrijving opgesteld. Deze leerstofomschrijvingen per vak bevatten tevens een opsomming van de daarmee samenhangende vakken, die afgeleid is uit de interactieschema's. De docent die het vak gaat verzorgen is verantwoordelijk voor het gestalte geven van die samenhang, wat tot uiting komt in gekozen notaties, voorbeelden, zichtbaar maken van verbanden en het realiseren van een vloeiende overgang van voorafgaande onderwerpen naar nog komende onderwerpen.

2.8 De implementatie

Het resultaat van paragraaf 2.7 bestaat uit vijf stromen van vakken, corresponderend met de vijf gebieden. Bij het verdelen van die vakken over trimesters spelen de volgende aspecten een belangrijke rol:
- het oriënterend karakter van het eerste jaar

- het selecterend karakter van het eerste jaar
- een homogene verdeling van de studielast
- voldoende praktische vaardigheid in het basisdeel
- onderwijsvormen: veel begeleiding in het eerste jaar, mondeling en schriftelijk rapporteren.

Door uit te gaan van de gedetailleerde vakomschrijvingen uit de voorgaande fase kon op eenvoudige wijze een verdeling over trimesters worden gemaakt. De vakken werden zodanig gerangschikt, dat niet meer dan vijf vakken per trimester ontstonden en bovendien de volgorde-afhankelijkheid werd gerespecteerd. Ten behoeve van de praktische vaardigheden werden praktica toegevoegd. Het zevende trimester is grotendeels gewijd aan twee grote opdrachten op de gebieden "Programmeren" en "Computerarchitectuur en Systeemprogrammatuur". Deze opdrachten dienen voornamelijk om de in de eerste zes trimesters opgedane kennis en vaardigheden te consolideren en te integreren. Het resultaat van de vakverdeling over trimesters is weergegeven in Figuur 2.

3 Onderwerpen per deelgebied

3.1 Abstracte automaten en formele talen:

- classificatie van formele talen en bijbehorende automaten
- berekenbaarheid, halting problem
- complexiteit

Motivering
- speelt een rol bij het modelleren van informatiesystemen, definitie van programmeertalen, compiler-instructie en parallellisme
- geeft de mogelijkheden en beperkingen van berekeningsmodellen.

3.2 Logica:

- propositielogica en eerste-orde predicatenlogica
- geldigheid, bewijsbaarheid, soundness en completeness
- andere formele systemen, zoals predicatencalculus en modale logica, datamodellen
- lambda calculus
- recursie en inductie.

Motivering
- Predicatencalculus vormt het gereedschap bij programmeren. Propositielogica en predicatenlogica vormen voorbeelden van logische systemen en modellen als opstap naar bijvoorbeeld Hoare-logica en temporele logica.

: 9 middagen instructie of practicum

progr. 0 Pascal 1.1 0 tu	wiskunde basiswiskunde 2.1 4 tu	wiskunde verzamelingen 2.2 2 tu	theor. informatica logica 1 3.1 3 tu	casp digitale technieken 4.1 3 tu	P colloquium
progr. 1 guarded command language 1.2 2 tu	wiskunde basiswiskunde 2.1 4 tu	wiskunde algebra 1 2.3 2 tu	wiskunde discrete wisk. 1 2.6 2 tu	casp micro-architectuur 4.2 3 tu	
progr. 2 guarded command language 1.3 2 tu	wiskunde basiswiskunde 2.1 4 tu	wiskunde algebra 2 2.4 2 tu	theor. informatica formele talen 3.2 2 tu	casp inl. comp. systemen 4.3 2 tu	
progr. 3 Pascal 1.4 2 tu	wiskunde traliethеorie 2.5 2 tu	W & MW 6.1 2 tu	theor. informatica logica 2 3.3 3 tu	casp implementatie 4.4 2 tu	D colloquium
progr. 4 abstr. datatypen 1.5 3 tu	progr. 5 parallel progr. 1.6 2 tu	wiskunde discrete wisk. 2 2.7 2 tu	theor. informatica prog. verificatie 3.4 2 tu	casp compilers 4.5 2 tu	
progr. 6 standaardalg. 1.7 3 tu	wiskunde kansrekening 2.8 2 tu	theor. informatica complexiteit 3.5 2 tu	theor. informatica logica 3 3.6 2 tu	casp operating systems 4.6 2 tu	
progr. 7 opdracht 1.8 0 tu	informatiesystemen modelbouw 5.1 4 tu	theor. informatica semantiek 3.7 2 tu	casp aspecten van progr. talen 4.8 2 tu	casp opdracht 4.7 0 tu	

Figuur 2
Vakverdeling over trimesters

- Recursie en inductie spelen een rol in tal van informaticagebieden zoals abstracte datatypen, semantiek van repetitie en procedure, automaten en talen en de theorie van berekenbaarheid. Daarnaast bieden zij inzicht in ingewikkelder structuren en daarbij behorende bewijsmethoden.

3.3 Syntax en semantiek

- syntax van programmeertalen
- Hoare-logica en varianten; partiële en totale correctheid
- semantiek; bewijsbaarheid en geldigheid van programmalogica's.

Motivering
- Levert de theorie die ten grondslag ligt aan de methoden en technieken die bij het ontwerpen van algoritmen worden gebruikt.

3.4 Aspecten van programmeertalen

- Enkele specifieke programmeertaalconstructies (zoals parameter-mechanismen, datatypen, control structures) en hun betekenis.

Motivering
- Levert criteria voor het beoordelen van bestaande programmeertalen en taalconstructies.

3.5 Wiskunde als ondersteuning van de informatica

- intuïtieve verzamelingsleer, relaties, ordeningen, (partiële) functies, transitieve afsluiting
- algebra: monoïden, groepen, isomorfiestellingen, ringen, idealen, lichamen, algebra's
- tralietheorie: c.p.o., volledig tralie, monotonie, continuïteit, dekpuntsstellingen, dekpunts-inductie, functieruimten, cartesische produkten, composities.

Motivering
De eerste twee onderwerpen komen op tal van plaatsen aan de orde:
informatiesystemen, automatentheorie, programmeren etc..Tralietheorie speelt met name een rol in denotationele semantiek en parallellisme. Verder voorzien deze vakken in de nodige bewijsvaardigheid, het leren omgaan met formalismen en het kunnen abstraheren van specifieke instanties van bepaalde structuren.

3.6 Wiskunde als algemene basis

- discrete wiskunde: tellen, eenvoudige getaltheorie,

grafen, combinatoriek.
- analyse, gericht op technieken en vaardigheden: technieken van de differentiaal- en integraalrekening, reeksen, complexe getallen, lineaire differentiaalvergelijkingen, partiële afgeleiden (niveaulijnen, gradiënt).
- lineaire algebra, gericht op technieken en vaardigheden:
  lijnen en vlakken, inwendig product en vectorieel product, lineaire afbeeldingen, matrices, stelsels vergelijkingen, determinanten, LU-decompositie, coordinatentransformatie, eigenwaarden en eigenvectoren.
- numerieke wiskunde, enige kennis van machinaal rekenen met name ten aanzien van reële getallen: nulpuntsbepalingen, oplossen van stelsels vergelijkingen, elementaire numerieke integratie.
- kansrekening en statistiek: kansrekening, discrete en continue stochastische variabelen, standaardverdelingen, momenten, wetten van grote aantallen, centrale limietstelling.

Motivering
Tal van problemen kunnen worden opgelost met behulp van elementaire technieken en methoden. Deze vakken voorzien hierin. Tevens past eenvoudige modelvorming binnen de behandeling van de genoemde onderwerpen.

3.7 Programmeren

- praktische vaardigheden met betrekking tot het verwerken van programma's
- programmaconstructies met bewijsregels (skip, toekenning, selectie en repetitie, concatenatie); specificaties
- programmeertechnieken (opstellen van invarianten, stepwise refinement, abstractie)
- datatypen en datastructuren
- standaardalgoritmen en standaardtechnieken zoals dynamisch programmeren, divide & conquer, backtracking, programmatransformaties
- procedures en functies
- parallelle programma's met gemeenschappelijk geheugen (P,V-operaties).

3.8 Computerarchitectuur en systeemprogrammatuur

- digitale bouwstenen: combinatorische schakelingen, registers, ALU.
- micro-architectuur: microbesturing, instructiedecodering, datapaden.
- machine-architectuur: registermachines, stapelma-

chines, adresseringswijzen, programmeren in machinetaal, parallelle en dataflow machines.
- compilers: afbeelding van brontaal naar doeltaal, globale structuur; lexicale, context-vrije en contextafhankelijke analyse; foutendetectie; codegeneratie.
- besturingssystemen: met name procesbeheer en geheugenbeheer.

3.9 Informatiesystemen

- conceptueel ontwerp van informatiesystemen, automaatmodellen, datamodellen, specificaties van constraints, queries en updates.

4 Evaluatie

In 1989 is het nieuwe curriculum voor het basisdeel van start gegaan. Om te bepalen in hoeverre dit aan zijn doel beantwoordt is een commissie "Begeleiding nieuwe curriculum" ingesteld. Deze commissie heeft als taken:
- opsporen van knelpunten in het curriculum
- beoordelen van lesmateriaal
- onderzoeken of de gewenste samenhang wordt gerealiseerd
- de voortgang van het eerste en tweede studiejaar bewaken.

Hiertoe wordt na elk trimester een enquête onder de studenten en de docenten gehouden. Tevens beschikt de commissie over de literatuur die bij de colleges en praktica wordt gebruikt. Het is nog te vroeg om nu uitspraken over het curriculum te kunnen doen. Wel is het zo dat het materiaal dat geleid heeft tot dit curriculum een goed houvast biedt voor een evaluatie. Ook correcties op het curriculum zullen op grond van de uitgangspunten worden uitgevoerd.

5 Afsluiting

De in dit artikel geschetste methode van curriculumontwerp heeft een aantal goede eigenschap-pen, waarvan sommige kenmerkend zijn voor top-down ontwerp in het algemeen:
- Het ontwerpproces bestaat uit een aantal stadia, waarvan steeds het resultaat is bediscussieerd, bijgesteld en gefixeerd alvorens verder te gaan met het volgende
  stadium. Door deze werkwijze is het duidelijk waar ontwerpbeslissingen worden genomen en wordt verstrengeling van overwegingen van verschillende ni-

veaus vermeden.
- Doordat de verschillende stadia in brede kring besproken en becommentarieerd zijn, heeft het curriculum een breed draagvlak.
- Door de gevolgde methode van stapsgewijze verfijning heeft het eindproduct een duidelijk herkenbare structuur. In het bijzonder is de samenhang tussen vakken expliciet gemaakt en welomschreven.
- De duidelijke structuur vereenvoudigt ook toetsing en evaluatie van de concrete invulling van het curriculum.

Gezien deze resultaten mag geconcludeerd worden dat de gevolgde methode van stapsgewijze verfijning voor curriculumontwerp goed bruikbaar is.

Noot

[1] Een aantal collega's binnen de vakgroep Informatica van de TUE hebben een belangrijke rol gespeeld in het totstandkomen van de beschreven wijze van werken. Met name willen wij prof.dr. F.E.J. Kruseman Aretz en dr. R.P. Nederpelt voor hun bijdrage bedanken. Voor het vervaardigen van de figuren bedanken we dr.ir. G. Zwaan. Tenslotte zijn we Margret Philips erkentelijk voor het verzorgde typewerk.