

Elementen ener afstudeerrichting in de informatica.

In de ciro is diepgaand gediscussieerd over de vraag welke stof niet zou misstaan in het curriculum van een wiskundig ingenieur die zich tot "software engineer" zou willen ontwikkelen. Wij zijn uiteindelijk terecht gekomen bij acht "verplichte vakken" -verplicht in de dubbele zin, dat elke toekomstige software engineer ze redelijk zou moeten beheersen en wij ze regelmatig zouden moeten geven- en zes "keuzevakken"; dit laatste zestal is niet uitputtend en ook minder definitief, het is meer bedoeld als aanduiding van het soort dingen, die b.v. voor een caput-college in aanmerking zou komen.

Wij geven de verplichte vakken hieronder in een min of meer chronologische volgorde.

1. Inleiding tot de Kunst van het Programmeren (2 \* 2 semesteruur).

Dit zij een college met een sterk methodologische inslag, waarin kennisoverdracht een vrij ondergeschikte rol speelt. De nodige semantiek wordt intuïtief gepresenteerd, de zorg om correctheid fungeert voornamelijk als richtsnoer bij het proces van programmaontwerp. In zijn huidige vorm beslaat dit college 1 \* 2 semesteruur; de voortzetting dient meer aandacht te schenken aan het gebruik van representatieve abstractie en de studenten indien mogelijk met omvangrijkere programmeeropgaven te confronteren.

2. Toegepaste logica (2 \* 2 semesteruur).

De bedoeling van dit college is om de student vertrouwd te maken met een aantal formele technieken van standing, welke onmisbaar zijn als men het vak op een solidere basis wil bedrijven. Het eerste semester is inmiddels geconsolideerd en bestrijkt twee hoofdgebieden, ten eerste formele systemen en ten tweede de beslisbaarheidsproblematiek. Het tweede semester is nog niet zover geconsolideerd; het college heeft bestreken de eindige automaat, generatieve grammatica's, ambiguïteit en herkenning met al of niet eindige automaten. De verwachting bestaat dat in dit laatste gedeelte een accentverschuiving van Chomsky naar Church zal plaatsvinden.

3. Inbeddingsproblematiek (1 \* 2 semesteruur).

Dit zij een college met een veel meer encyclopaedische inslag, waarin de adresseringsproblematiek en technieken om deze problematiek op te lossen centraal staan. Onderwerpen als onafhankelijke vertaling, segmentering, linking, gemeenschappelijke bibliotheek, stacks, displays, co-routines e.d. dienen hier aan de orde te komen. Deze stof wordt thans gedeeltelijk gedekt door het huidige college Informatica I en het college Operating Systems. De bedoeling van dit college is om een gedeelte van de kennis en de begrippen aan te dragen om een computer systeem op zijn adequaatheid te taxeren. Dit college moet nog worden opgezet, het zal heel moeilijk zijn de stof op het vereiste niveau en voldoende samenhangend te presenteren.

4. Representatietechnieken (1 \* 2 semesteruur).

De bedoeling van dit college is om de studenten vertrouwd te maken met de diverse technieken van afbeelding van abstracte informatiestructuren (strings, matrices, lijsten, grafen, meer-dimensionale objecten, naamlijsten e.d.) op geheugenstructuren, zoals linked lists, hash-coding etc. Deze stof wordt thans gedeeltelijk bestreken door de colleges Informatica II en III. Omdat dit college Representatietechnieken tamelijk encyclopaedisch gedacht is, lijkt bij nader inzien de kans op overlapping met de voortzetting van de Inleiding tot de Kunst

van het Programmeren aanmerkelijk kleiner dan we aanvankelijk even dachten. Het zal wederom heel moeilijk zijn de stof op het vereiste niveau en voldoende samenhangend te presenteren.

#### 5. Syntactische Technieken (1 \* 2 semesteruur).

De bedoeling van dit college is om de student te leren zijn talen zo te ontwerpen, dat diverse aspecten van de mechanische tekstverwerking (parsing, generating, "back-on-the-rails", type-checking) gemakkelijk mogelijk worden. De terugkoppeling op taalontwerp zij het hoofdmotief van dit college. Het is nadrukkelijk de bedoeling de uitgebreide catalogus van bestaande parsing technieken (meestal voor talen, die nooit ontworpen hadden moeten worden) met dit college buiten de deur te houden. Dit college zal geheel nieuw moeten worden opgezet, over de bereikbare kwaliteit van dit college maken wij ons niet de meeste zorgen, het is evenwel denkbaar, dat de stof zich laat reduceren, zodat het voor 2 semesteruur wat te weinig wordt.

#### 6. Axiomatisering van Semantiek (1 \* 2 semesteruur).

De bedoeling van dit college is

- 1) studenten vertrouwd te maken met axiomatische definitie van de semantiek van programmeertalen (ten gunste van mechanistische)
- 2) allerlei bruikbare stellingen -die in de Inleiding tot de Kunst van het Programmeren intuïtief ten tonele gevoerd zijn- nu rigoureuus af te leiden
- 3) waar mogelijk de consequenties van het voorafgaande ten aanzien van taalontwerp te trekken.

Vergeleken met wat we nu doen is dit een helemaal nieuw college, maar we verwachten, dat er over een a twee jaar voldoende stof voor is.

#### 7. Informatiebeheer (1 \* 2 semesteruur).

De bedoeling is de studenten vertrouwd te maken met de moeilijkheden, die bij het bespelen van grote gegevensbestanden optreden en met de methoden die worden toegepast om aan deze moeilijkheden het hoofd te bieden. Wij zouden erg graag zien, dat in dit kader aan "error recovery" op een meer systematische manier aandacht zou worden geschonken, dan wij tot op heden kunnen. Wij verwachten helaas niet, dat aan deze vrome wens binnen een paar jaar voldaan kan worden. Dit college beschouwen wij als de continuering, resp. ontwikkeling van het huidige college Informatica II.

#### 8. Parallellisme (1 \* 2 semesteruur).

Dit college beschouwen wij als de continuering, resp. ontwikkeling van het huidige college Operating Systems. Doordat een gedeelte van de thans descriptieve functie van dit college door het college "Inbeddingsproblematiek" wordt overgenomen, wordt er ruimte geschapen voor de behandeling van een ruimere categorie synchronisatieproblemen, b.v. tussen via boodschappen communicerende machines.

Als onderwerpen voor keuze colleges zijn genoemd LISP, SNOBOL, Petri netten, Complexity Theory, Ontwerpmethodologie, Bootstrapping en Informatietheorie.