

Reisverslag van bezoek aan het Seminar on the Teaching of Programming at University Level.

De vier hoofdsprekers waren (involgorde van optreden) ondergetekende (4 uur), Knuth, Iverson en Perlis (alle 3 uur). Deze optredens waren allemaal te lang, te vermoeiend, zowel voor de spreker als voor het gehoor en een van de dingen, die ik me van dit seminar zal blijven herinneren is, dat dit geen goed arrangement is. (Ieder van ons begon met twee uur ononderbroken!) De organisatoren lieten aan de sprekers het organiseren van discussies over; ook dit was niet gelukkig.

Mijn verhaal "The Art of Programming" liep redelijk, op een paar inzinkingen na; en ik heb te weinig expliciet ruimte voor discussie gelaten. Ik had in het begin wel gezegd, dat ik interrupties op prijs stelde, maar dat bleek onvoldoende. (Zo verging het de anderen ook.)

Don Knuth viel me wat tegen; ik weet nu echt, dat ik zijn encyclopaedie niet hoeft te bestuderen. Die heet "The Art of Programming", de titel van zijn lezing was "The Analysis of Algorithms" en dat is een veel adequatere titel. Hij heeft de instelling van een non-toegepast wiskundige: computing science is voor hem voornamelijk een onuitputbare bron voor zijn puzzletjes. Het is heel merkwaardig, misschien is het erg Amerikaans: eerst geven ze computing science de meest vulgaire motivering (minimizing resource usage), en vervolgens keren ze zich er met enige walging van af ~~XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX~~ "to do the really interesting problems". Zo vind je Don Knuth trots de variantie bepalen van het aantal gegenereerde carries bij een optelling in het b-tallig stelsel, terwijl het enige belangrijke is de lengte van de maximale carry-trein!

Iverson was walgelijk, dit was een "slick commercial" over APL. Het weer-~~XXXXXXXXXX~~ zinwekkendst vond ik zijn grofheid jegens zijn publiek. Als het zijn bedoeling is geweest om APL te verkopen, dan is hij hierdoor niet geslaagd: het publiek accepteerde hem niet. Ik kan het ontwerp geen consistentie ontzeggen (zal dat ook niet doen), wat op het ~~XX~~ eerste gezicht alleen maar ad-hocery lijkt, blijkt meestal op een of andere manier overwogen. Het heeft mijn indruk bevestigd dat de invloed van APL op zijn gebruikers funest is: it is a giant bag of tricks! En inplaats van dat het de neiging programmeren te zien als puzzlen bestrijdt, moedigt het gereedschap deze opvatting aan. Ik dacht -en geloof dit nog- dat programmeren in de eerste plaats een conceptuele uitdaging is en zie niet, hoe APL in dit opzicht een stap vooruit is. Het treft mij zelfs als een stap achteruit.

Perlis was bijzonder aanhoorbaar. Zijn inleiding vond ik erg verhelderend, waarin hij betoogde, dat "programma correctheid" niet los gezien kon worden van wat het programma doen moest, daardoor onverbrekelijk ~~XXX~~ verbonden was met ons theoretisch inzicht in de toepassing! De intellectuele uitdaging, die het bestaan van computers inhoudt, wordt daarmee tot de behoefte aan fatsoenlijke theorieën, over activiteiten, die we tot nog toe intuïtief, op zijn Jan-Boere-fluitjes of niet deden. Het overtuigendst vond ik de beschrijving van het eerste college (hij beschreef er drie), waarin hij "from top to bottom" de relatie mens → algoritme → taal → machine → mens beschreef. Eerst programmeren in een hogere programmeertaal en dan, naarmate de verwerking van de algoritmen gedetailleerder behandeld wordt, de taal~~X~~ van zijn franje ontdoen. In zijn ~~XXX~~ tweede college vergeleek hij verschillende programmeertalen, in het derde besprak hij kwaliteitsaspecten van programmeertalen. Het streven was nobel, maar dit laatste vond ik minder overtuigend, te meer, doordat hij zich kennelijk nogal gebonden achtte aan wat er was.

Twee sprekers (elk van een Engels software house) hebben de praktijk van het programmeren in de industriële omgeving op hun eigen manier belicht. d'Agapeyeff streek het gehoor erg tegen de haren door te berogen, dat hij aan de computer scientists zoals ze van de universiteit kwamen, zo weinig had. Hij stelde daar zijn eigen, interne opleidingsprogramma tegenover; de algemene reactie was, dat het een voortreffelijke taakverdeling was, dat hij zijn eigen mensen dat dan maar moest leren, dat dit soort opleiding onmogelijk op de weg van de universiteiten kon liggen. De vraag werd zelfs gesteld, of hij voor dit soort werk wel "graduates" nodig had. Door zijn overdrijving kwam deze discussie niet goed op gang. Jackson liet zien, wat in zijn bedrijf gebeurde; dit was een warm pleidooi voor structurering, dat bij het publiek beter in de smaak viel. (Wat natuurlijk ook aan het publiek lag!)

J.Aron (Federal Systems Division, IBM) hield een verhaal in de terminologie van de manager. Mensen, die dit nooit hadden gehoord of niet wilden horen, werden er erg boos over. Ik persoonlijk heb een zwak voor deze man, ik heb me niet geergerd en er zelfs wat van geleerd: FDS probeert om van de "Chinese Army Approach" af te komen. Of ze daarin succes zullen boeken, lijkt erg de vraag, aangezien de behoefte aan the Chinese Army voor een niet gering gedeelte gedictieerd wordt door de barokheid van het gereedschap. Het was boeiend te horen, dat ze er van af wilden, tot een jaar geleden konden ze ons met reden omkleed ~~XX~~ aantonen, dat the Chinese Army de enige mogelijk was!

Wat~~X~~ de "gemiddelde" deelnemer van dit seminar heeft opgestoken, in welke richting zijn denken gestimuleerd is, vermag ik niet te taxeren, het is voor mijzelf zelfs erg de vraag, of, en zo ja, hoe, ik er als een wijzer mens vandaan ben gekomen. Een aantal mensen hebben op de laatste middag samenvattingen van hun indruk gegeven (van der Sluis, Hoare, Wirth, Samelson en Michaelson). Zelfs als ik er van af trek, dat ze daar stonden in een positie die er om vraagt om vriendelijke dingen te zeggen, leek het toch, of ik het er niet zo slecht van af gebracht had. Mij viel op, dat ik unaniem als "artist" geclassificeerd werd; ik vraag me wel af, wat ze er mee doen en wat ik daar mee doe! Het concept van "Structured Programming" slaat erg aan, het lijkt voor mij noodzakelijk om in deze stijl nu eens iets groots aan te pakken, maar ik ben daar wel bang voor. Er moeten dan een heleboel detailbeslissingen genomen worden en tot nog toe heb ik voornamelijk aangetoond, dat deze allemaal onverwacht veel zorg vergen.

Edsger W.Dijkstra