

Het einde van een ambacht.

In de wijze waarop met rekenautomaten wordt omgesprongen begint een drastische verandering zich af te tekenen en in verband met de consequenties hiervan lijkt het gunst deze verandering tijdig te signaleren. Kortweg kan men stellen, dat het programmeren het ambachtelijk stadium ontgroeit tot een wel-gefundeerde, tamelijk wiskundig gerichte discipline.

Een en ander is een direct gevolg van de capaciteitsvergroting van het technische vermogen, zoals wij deze in het laatste decennium aanschouwd hebben. Enerzijds heeft deze vergroting geleid tot een afzwakking van de efficiencyeisen, gemeten naar de wat eenzijdige maatstaven van geheugenruimte en rekestijd, zodat het denken daardoor nu minder verlamd is, anderzijds heeft deze vergroting toepassing gebieden ontsloten, waarbij ⁿdu blijkt dat het ongetrainde menselijke bevattingsvermogen verreweg het nijpendste knelpunt is geworden.

Het eerste teken aan de wand was het omhoogschietende budget, dat voor software-~~XXXXXXXXXX~~projecten ter beschikking gesteld moest worden, een ontwikkeling die aanvankelijk met enig succes door organisatietechniek nog wat is afgeremd. De noodklok is pas goed gaan luiden toen ~~XX~~ een groot gedeelte van de aldus voor grof geld vervaardigde software na aflevering bleek gewoon niet te deugen.

Dit falen was een intellectuele uitdaging, die mede vanwege de economische belangen niet onbeantwoord kon blijven en dan ook niet onbeantwoord gebleven is. Bewijstechnieken, waarmee de correctheid van zowel sequentiele programma's als ook van multiprogrammeringsystemen kan worden aangetoond, zijn en worden ontwikkeld, bewijstechnieken, die niet zullen nalaten de stijl van programmeren diepgaand te beïnvloeden: een nauwkeurig aanvoelen van de factoren, waarvan de hanteerbaarheid van dergelijke bewijzen kritisch kan afhangen, zal binnen niet te lange tijd tot de vereiste vaardigheden van de competente programmeur gerekend moeten worden. Bovendien: als men met de opbouw van zijn programma er rekening mee houdt, dan blijkt de opgeve om een serie testgevallen te construeren, waarmee de juistheid van een programma aan de machine uitputtend kan worden vastgesteld, helemaal niet zo onmogelijk als dikwijls nog wordt aangenomen.

Heden is het nog niet ongebruikelijk, dat grote programma's min of meer op de bonnefooi gemaakt worden en dan aan enige vrij willekeurig gekozen testgevallen worden onderworpen om althans de grofste blunders aan het licht te brengen en -zij het vaak ten koste van veel machinetijd- te localiseren. En zulke programma's gaan daarna als "geteste programma's" de wereld in! Maar het moment lijkt nabij, waarop een dergelijke manier van werken met recht dilettantisme genoemd mag en ook zal worden.

Van de professionele programmeur van zo niet morgen, dan toch ~~de~~ overmorgen, zal aan abstractievermogen -nodig om structuur te onderkennen en beheersen- en aan mathematische bewijsvaardigheid aanmerkelijk meer gevraagd worden dan ooit redelijkerwijs gevergd kan worden van een jongeman, wiens geestelijke bagage niet meer omvat dan een programmeurscursus van een paar maanden met daarna een praktijk van een paar jaar. Hoe snel de verandering in rekenklimaat zich zal inzetten, laat zich moeilijk voorspellen: tegenover de economische pressie werk van een klasse beter (goed nl.) af te leveren staat de remmende invloed van het in vele instituten overvloedig aanwezige kader van middelbaar niveau of lager, waarvan we niet mogen verwachten dat het, zo het zulks al ziet, gaarne erkennen zal dat het vak boven zijn geestelijk plafond is uitgegroeid, dat zijn ambacht heeft opgehouden als zodanig te bestaan.

Prognoses over het aantal rekenmachines lijken me wat klakkeloos vertaald in prognoses over de behoefte aan programmeurs van het thans gangbare niveau. Is het nog langer verantwoord middelmatige jongens met niet meer dan middelbare school naar programmeurscursussen te lokken met de suggestie, dat ze daarna onmisbare specialisten zullen zijn, als we zien aankomen, dat ze ~~het~~ door de bank genomen ^{professioneel niet mee zullen kunnen?} niet verder zullen brengen dan professioneel ~~uitse~~ ^{uitse}? Het is in dit verband tekenend, dat een Angelsaksische instelling, naar ik onlangs vernam, voor programmeurs met een Bachelor's Degree geen genoeg meer neemt: Master's Degree is een minimum, Ph.D.'s hebben ze liever.

prof.dr.E.W.Dijkstra

Technische Hogeschool Eindhoven