Trip report E.W.Dijkstra, Edinburgh and Newcastle, 1 - 6 September 1974.

On Sunday 1st September 1974 I flew via London from Amsterdam to Edinburgh, where I arrived with a delay of only two hours. The little bit of the afternoon that was left and the evening I spent talking shop with Rod Burstall, who was my host. Mondaymorning I passed at the Department of Machine Intelligence, Hope Park Square, where we were joined by a number of staff members of the Department of Computer Science. The somewhat erratic nature of that morning's discussion seemed to reflect the way in which the various related groups of the University are scattered all over the town. Carl Hewitt from MIT had to devide his attention between the subject of the conversation and one of Edinburgh's female students, whom he found at least as attractive. There was also Richard Waldinger from Stanford Research Institute, who confirmed my impression of SRI as had been formed a week before, when I had received a visit from two other people from that institute. Waldinger referred to that visit, but in order not to spoil the atmosphere I did not pursue that topic. The more I hear about Artificial Intelligence, the more ridiculous becomes its often heard defense that Artificial Intelligence has contributed so much to programming technology --for both, the claim and its debunking, see the various contributions to the Lighthill Report--. I see more and more reason to characterize its contribution as "adding to the confusion".

LISP is a good example. In the early sixties we have suffered from a piece of computing science folklore, viz. that the crucial test whether one had designed "a good language" was, whether one could write its own interpreter in itself. I have never understood the reason for that criterion --it was suggested that such a language definition would solve the problem of the definition of (particularly) the semantics of the language--, the origin of the criterion was the example of LISP, which was "defined" by a ten-line interpreter written in LISP, and that, somehow, set the standard. (I found this hard to accept, because when I started to study LISP from the LISP 1.5 Manual, I could not understand it: after an incomplete definition of the language, its authors try to supplement these deficiencies by an equally incomplete sketch of a specific implementation, and that was a confusion if there ever was one!) This unfortunate LISP-interpreter formulated in LISP has somehow given mechanistic (or "operational") language definitions an undeserved aura of respectability, and its unwholesome influence can be traced right through the Vienna Definition Language and ALGOL 68 --what I knew-- and even --what I learned this week to my surprise and disappointment-- to the work of Dana S.Scott. It is enlightening to know --what I also learned later that week-- that LISP's most awful property, viz. "shallow binding" --now religiously defended by its addicts!-- is described by McCarthy as "a mistake" in that ten-line interpreter (which, of course, it is)!

On Monday afternoon I gave a formal lecture in which I showed the development of Rem's Algorithm for the Recording of Equivalence Classes, and I found it an uncongenial audience. Immediately afterwards I sought the explanation in the AI-environment to which designing something that should meet rigourous specifications and working backwards from them does not come naturally. It was only later that week, while in Newcastle, that I realized that in a sense I had been "too modern" for them, and that I had underestimated the extent to which the mechanistic approach was still in full vigour. The next day Rod Burstall told me that a number of people had told him that --contrary to my impression-- they had liked the talk very much. Early in the evening I went by train from Edinburgh to Newcastle: there was a real Scottish rain outside and I had plenty of time to sort out my mixed feelings.

The next four days I attended at the University of Newcastle the yearly seminar, this time on "Formal Aspects of Computing Science." (My sweater was identified as one of the informal aspects of computing science.) This was an exhausting affair because --in contrast to previous seminars-- all speakers were of such a calibre that one did not want to miss a single lecture.

Three speakers dealt with various aspects of the subject now known as "computational complexity" and among themselves --or by accident, but I don't think so-- they had divided the subject quite nicely.

Dr.M.Rabin --rector of the Hebrew University of Jeruzalem-- stole the show as far as I was concerned. He was a very capable --and probably very experienced-- lecturer, who mainly gace results (and definitions, needed to formulate these results) but hardly proofs, sometimes a sketch of a proof. While the distinction between decidable and undecidable is now quite well-known, Rabin dealt with "decidable but.....", e.g. decidable theories where the number of steps required for the shortest possible proof of a theorem could explode (super)exponentially with the length of the theorem (and we don't even talk about the search process usually neede for discovering the proof). A sobering thought. I had not studied the program of the seminar at all and Rabin's collaboration was therefore a surprise for me, a most exciting one even. (I had already been thrilled by his IFIP paper.)

Dr.S.Winograd of the IBM Research Center in Yorktown Heights was the second speaker on the same subject. Apparently Rabin and Winograd knew each other fairly well, they also resembled each other in their way of presentation: slightly mocking and irreverent, and very modest. (This is perhaps a consequence of the subject!) Winograd follwed the same pattern of showing results and sometimes sketching a proof, but Winograd focussed our attention on the minimum number of operations needed for (otherwise quite finite) problems as matrix multiplication etc. Being less easily excited about efficiency considerations than many others (Knuth!), I was quite surprised to find my attention fully absorbed by Winograd's lectures. (Was it because I recognized some of the tricks I had played myself in the late fifties when making subroutines for standard functions?) Both Winograd and Rabin struck me as "wise men".

The third speaker on complexity was professor John Hopcroft from Cornell University, who showed that from the point of view of complexity many at first sight different problems can be viewed as different instances of the same problem, with the result that superficially different, shrewd algorithms can be view as instances of the same method. I had met Hopcroft, but had never heard him lecture: it was a pleasure and I shall buy his new book. I had heard him state his opinion that, essentially, there are no more than, say, thirty basic algorithms and that today a competent computing scientist should know them: the days of the bright young lad without the necessary knowledge are past. He made his case so convincingly that I started wondering to what extent the book I am writing --mostly without the aforementioned knowledge-- will be obsolete even before the printer's ink has dried. And this my worry is saying a good deal about the quality of Hopcroft's lectures! I think that "unifying and consolidating" are about the most appropriate terms for the contents of his message.

The fourth speaker that I had never heard before was professor D.S.Scott, now from Oxford University. His first lecture was a bitter disappointment. He reacted rather inadequately when asked for clarification about terminology, notation and even inconsistency displayed by his prepared transparencies. With the possible exception of his faithful followers --for whom we cannot blame him-- he left his audience very unsatisfied. In the discussion at the end of the first day our host, professor E.S.Page from Newcastle, urged him to use his next slot to try to make at least his purpose clear. So he tried to do the next day (his effort to do so was a great relief for most of us: it showed his good will and that his previous failure to communicate had not been caused by contempt for his audience). Unaware of what had happened two years ago to speakers who did the same, he explained his purpose in the good American tradition via a diagram with arrows. His audience appreciated his effort and, in order to avoid further embarrassment, refrained from asking what the arrows meant. (I needed at least two different meanings, probably more....) On the whole he struck me as the mathematician who, in order to help the English bookkeeper, develops the theory of mixed radix number representations several years after Brittain has switched to decimal currency. His last lecture, which had very little connection with the previous ones (nor with computing science, for that matter) was self-contained and much better. It had very much the flavour of John Backus's Reduction Languages and van der Poel's fiddling with the so-called paradoxical combinator. Although (or because?) I do not care too much about that subject, I could stand that last lecture very well.

The fifth lecturer was dr.H.Bekic from the IBM Laboratory in Vienna on "Semantics of Parallel Programs". In his first lecture he was so shy and nervous that it was hard to see what he was driving at. In his next lectures this became clearer but the effort struck as misguided in more than one sense. In a very mechanistic approach he tried to formalize the non-determinacy caused by the arbitrary interleaving of (the executions of) arbitrary programs, situations no one cares about (or should care about after three minutes reflection). Secondly he tried to squeeze his formalization into Scott's formalism. I pitied Scott, for the horrible formulae produced by Bekic could only act as a witness for the prosecution. Of course one can dismiss the whole thing by saying that, with VDL to formalize the definition of PL/I in its past, one can expect anything from that crew (which, of course, is true: the crew members themselves, I guess, will be the last ones to deny it). Looking at it from the outside, however, one cannot escape the uneasy feeling that that crew is under heavy company pressure to tackle the wrong problems. And the company in question is large enough to consider such ill-directed pressure as a rather alarming symptom of the general misunderstanding of the role that scientific thought could and should play in our culture. (I shudder at the thought of the havoc than can be caused by similarly ill-directed funding policies of our various national governments! Also there, frightening things could nowadays happen quite easily!)

Side remark. In passing Bekic mentioned that the group in Vienna now wrote semantic definitions "in a new style". At the end of that lecture he got an alarmed question about that new style from Ollongren: Ollongren had just published a book on VDL! In a microcosmos we could see how manufacturers must have felt when IBM changed from 7-track to 9-track magnetic tapes; it was very funny....

The sixth speaker was professor C.A.R.Hoare from the University of Belfast, who devoted three talks to data representation, monitors and parallel programming respectively. In spite of the fact that I had heard most of it at earlier occasions, it was, as always, a pleasure to listen to him. More or less expecting three disjoint talks. I was pleasantly surprised to see he tied them in together. He is a very cautious man, who tries to avoid the introduction of complications by not generalizing more wildly that he can cope with, instead of the immodest compounding of combinatorial complexity generators so many language designers seem to revel in. (It was Hoare, who stressed in North-Berwick in 1968 already the need for humility!) Another quotation: "Reliability can only be achieved by the utmost simplicity, but most manufacturers think this a price too high to pay!" Wednesday afternoon he and I did not join the excursion and talked together for four hours.

The opening lecture "Formalization, history, present and future" was by professor H.Zemanek from the IBM Laboratory in Vienna, a nice talk that suited the occasion. Our host, E.S.Page, had given a long quotation from von Neumann about mathematicians formalizing for formality's sake. I thought it a pity that at the closing session he did not repeat that quotation: those whom it would concern would probably fail to hear the warning. There were 56 participants and, as a result of the subject, there was a relatively heavy representation of what now pushes itself as "theoretical computing science" which, I am afraid, is just a euphemism for "unpractical computing science"; and the latter could very well turn out to be a contradictio in terminis. The split in the audience was rather marked. I saw Nivat (from the University of Paris) and found my impression confirmed that the new journal (!) "Theoretical Computer Science" is no more than the next political move of the French mathematicians, who seem to fear but one thing: to loose their power. I guess that they will succeed in preventing the birth of Computing Science in France for at least another decade and it is always nice to know of the existence of yet another journal one can ignore. I am willing to make a bet that de Bakker will publish in it....

Ollongren and de Bakker were my only countrymen at the Newcastle Seminar; Verrijn Stuart, who did attend previous seminars, was absent.

16th September 1974
Burroughs
Plataanstraat 5
NUENEN - 4565
The Netherlands

prof.dr.Edsger W.Dijkstra
Burroughs Research Fellow

cc.: Rabin
Winograd
Hopcroft
Scott
Bekic
Zemanek
Page
Randell
Burstall
Hoare