Trip report E.W.Dijkstra, Marktoberdorf, 29 July–
10 Aug 1986.

With speakers from half a dozen nations and
participants from two dozen, we formed a mixed lot;
observing the different national prejudices and inclina-
tions was both amusing and frightening. People
from (sub)cultures in which topic X is well-
established tended to show the attitude: "What
Computing Science is really about, is X.", and
for X you may substitute: predicate calculus,
typed λ-calculus and mathematical foundations,
logic, denotational semantics, category theory,
mechanization, or algebra. And, of course.
these views are all terribly incomplete and one-
sided, no matter how consistent each of them may
be in isolation. Personally I was relieved to
see that my own X –viz. "the mastery of com-
plexity" – is pretty broad. (That is probably why
it has served me reasonably well for more than
two decades.)

Besides these technical differences, the participants
had to cope with different educational philosophies;
and so had the lecturers, who were blamed for
their diversity. These differences were the more con-
fusing as they remained under the surface.

Crudely stated, the educational dilemma is whether
we are aiming for (A) maximum competence, or (B)
maximum number of graduates. In idealistic theory,
it need not be a dilemma –each student trained to
his ability, etc. –, in practice it is: if essential topics

or techniques are beyond the sophistication or intellectual energy of some students, view A blames those students (and tries to get them removed), while view B blames the topics (and tries to get them removed).

The dilemma surfaced in different ways. In the coach on our way back, from the Boarding School to Munich, Richard S. Bird, temporarily representing view B, wondered whether we had not been confusing by collectively exposing the participants to five different notations for concatenation: would not it have been better if we could have settled upon a single one? Having seen the damage done to students for whom LISP is the only vehicle towards formality, I found myself arguing that the danger of becoming "mathematically monolingual" was far worse.

It also surfaced in the different appreciations of Gentzen's "Natural Deduction", designed to model how people "actually conclude" and "therefore", according to Gentzen, most appropriate in the classroom: it appeals to the student's intuition, it comes quite naturally, etc. (From my mouth you'll never hear "intuitive" or "natural" used as recommendations.) It is quite possible that, half a century ago, Gentzen never envisaged large-scale applications of formal techniques. In any case he missed that a major virtue of formalisms can be to free us from the shackles of our native tongues and the reasoning habits enduced thereby. The message that in actual usage Natural Deduction is a by now well-documented pain in the neck was

not universally welcomed. [It could very well be that Natural Deduction has been used to try to make view B scientifically respectable.]

It also surfaced in the different judgements of the potential of formal techniques. Repeatedly we heard —declared as dogma— that the application of formal techniques was too laborious and tedious to be of any practical value. Typically, such noises came from people that had never tried to apply formal techniques in real earnest, and I got the impression that the dogma is also used as an excuse for not trying. They were the people that were not attracted at all —to quote E.T. Bell re formalization— by "the loss of the privilege of making avoidable mistakes". They were also the people that had the greatest difficulty in hearing high-technology's clearest message, viz. that designs now tend to be judged along a very simple binary scale: flawed or flawless.

The educational dilemma was manifest, and nobody, I observed, had swept it as far under the rug as the British. [The other day, back home in Austin, I read in the (British) weekly "Computing" of the indignation caused by the immoral suggestion of "merit pay". No wonder that that economy is in trouble.)

*   *   *

I noticed an interesting linguistic shift. A well-known form of rabid anti-intellectualism used to be betrayed within a few minutes by the speaker's appeal to "the real world". But no more! The real

world is no longer fashionable; it is now "out there". Apart from the change in terminology, nothing has changed: an appeal to "out there" betrays, again within a few minutes, the speaker's same rabid anti-intellectualism. (More is happening, but that I only have from hearsay: one of my academic colleagues was approached by people from the British industry because they wanted to know "what was going on in the real world". Are they beginning to suspect that the computer industry so far has been a world of make-belief?)

I also found confirmed that satire is very hard to practise, because, no matter how good you think you have been at it, it takes reality no time to overtake you.

Early during the Summer School we had a Grill Party at Auerberg. At my table we spent some time stringing buzzwords together so as to concoct the theme for the 1988 Summer School. So, when a few days later over a beer someone asked me whether the next Summer School's theme had been decided, I could give him an instant reply "Non-monotone belief-based conflict resolution.". One of the participants swallowed it hook, line, and sinker and declared that he was very keen on attending that Summer School as well........

In my speech after the Farewell Dinner I divulged among other things that I was in the process of founding "The International Institute for the Cultivation and Conservation of Artificial Complexity". Alas! The other day I read the announce-

4

ment of "Software Engineering '86", a conference organised by the British Computer Society and the Institute of Electrical Engineers in conjunction with the Alvey Directorate at the University of Southampton, 10th-12th September 1986. There will be four parallel streams:

Software Engineering for Information Processing
Formal Methods in Software Engineering
Software Engineering for Real Time Systems
Software Engineering for Artificial Applications.

(I am <u>not</u> kidding!)

*   *   *

The audience was at least as mixed as usual. In the beginning I congratulated ourselves because it took a remarkably short time to loosen them up and to get a discussion going. But I got disenchanted because the number of people participating in the discussions remained very small: we kept hearing the same voices and eventually the discussions became boring. The others one met during lunch, dinner and excursions; eventually I had to admit —am I aging?— that too large a fraction was dull or vulgar.

Of my seven lectures, two were failures (at the first slots in the morning after the Grill Party and the Farewell Dinner). I did predicate calculus, derived proofs about extreme solutions and explained Batcher's Baffler. I liked my proof derivations most, the audience preferred Batcher's Baffler.

Of the other speakers, I liked the lectures of Richard S. Bird most. He showed a calculus in which he derived all sorts of programs computing functions on sequences. Both his derivations of programs and his proofs of theorems were very calculational. The fascinating thing was that he could justify almost all steps with "There is really only one thing you can do." It was a very convincing demonstration of the heuristic value of formalization.

Tony Hoare was the most controlled speaker: his visuals were beautifully prepared and he knew exactly how many minutes each would take. He talked about his Communicating Sequential Processes and he cannot be blamed that I had seen most of it (several times). But I am beginning to have some doubts about his topic, doubts that were reinforced by the observation that he continues to talk very operationally. I am no longer sure that today the concept of a process deserves such a central position in our thinking.

The lectures by Manfred Broy and Gérard Huet were very similar in that their visuals were totally inadequate and that, during their first five lectures, they snowed us under. During their last lectures the general reaction was "Oh, was that what you were trying to talk about! Why did not you start at the end?". I hope and think that Huet's material can be cleaned up considerably: the outcome could be a very nice, clean, and small logic.

Cliff Jones and Rick Hehner were both clear and

6

articulate, and that was greatly appreciated. Both of them sometimes seemed to make too much of a point of too little; neither of them was entirely happy about his performance. Do they suffer from the wrong kind of students?

J. Alan Robinson was fun to listen too, but the way in which he talked about resolution clearly betrayed that it had been conceived 25 years ago. I intended to use the nine-hour train trip from Munich to Eindhoven to study his handouts and to come home understanding all about resolution. His text, however, was so unsatisfactory that I gave up by the time we approached the German-Dutch border.

Listening to Michel Sintzoff was a unique experience. His first talk contained more half-uttered hints at analogies than clear statements, later it became clear that the analogies were daring. He then became very graphical, which put me off.

About the remaining speakers I can report very little.

*   *
      *

For me the Summer School was instructive. It gave me a survey of what different groups are trying; personally I found most inspiration (and confirmation) in Bird's work.

By accident I picked up another example of

7

how people are "formed" by the instruments they use. Over a beer I posed a geometrical problem[t], for which a very elegant solution can be worked out without pen and paper. Huet, who has worked for the last ten years on mechanical proof verification, translated the demonstrandum into a long, clumsy formula full of goniometric functions and concluded with "And now it is decidable", confirming my fear that a lot of the manipulation delegated to machines is perfectly avoidable.

I was struck by the immaturity of the audience. One expects from budding scientists that they believe that science may serve a purpose, one expects from budding computing scientists that they believe that computers may serve a purpose. But I do not expect full acceptance of the dogma, that under all circumstances "computerization" is a great improvement: doubting the dogma was treated as heresy. The sore truth is that the audience had been brainwashed.

t) Consider k points equally spaced around the perimeter of a circle with radius $=1$ and the chords drawn from one of those points to the others. Show that the product of the lengths of those $k-1$ chords equals k.

17 Aug. 1986

prof. dr. Edsger W. Dijkstra
Department of Computer Sciences
The University of Texas at Austin
Austin, TX 78750 - 1188
United States of America