One up again for calculational programming

(an addition to our files)

In this note we deal with the following little programming exercise. Given integer arrays $x, y\,[0..N)$, $1 \le N$, construct a program to compute the value of

$$\langle \forall i,j : 0 \le i < N \wedge 0 \le j < N \wedge i \ne j : x.i \le y.j \rangle \ .$$

It was Jayadev Misra who conveyed the problem – through the medium of a table napkin – to Edsger W. Dijkstra, who solved it and, in turn, communicated the problem to me – by fax – .

We would very much like to invite the reader to try to solve the problem for himself, preferably without pencil and paper to start with.

* * *

When I started to think about a solution without the use of pencil and paper, some sort of chaos tended to prevail. Then, with the use of these devices, I quickly solved the problem by employing one of our standard calculational techniques. I informed Dijkstra that I did it, be it that I had broken the symmetry between x and y. He replied that his solution was symmetric.

And because sometimes I have other things to do, I left the problem alone but put it on the agenda of the next ETAC.

* * *

The ETAC, too, started to consider the problem without use of the whiteboard. How does one express gently, that the resulting discussion was not too convincing? Then I explained my calculational solution, while at the same time regretting the asymmetry.

After a while, Lex Bijlsma reminded us that in this kind of problem, where there are no monotonicities in x or y, it often[f] pays to replace <u>all</u> occurrences of one and the same constant by one and the same variable. Following this proposal, we replaced both occurrences of N in our target expression by variable n and investigated an increase of n by 1:

$$\langle \forall i,j : 0 \le i < n+1 \wedge 0 \le j < n+1 \wedge i \ne j : x.i \le y.j \rangle$$

$\equiv$ { split off $i = n$ }

$$\langle \forall i,j : 0 \le i < n \wedge 0 \le j < n+1 \wedge i \ne j : x.i \le y.j \rangle$$
$$\wedge$$
$$\langle \forall j : 0 \le j < n+1 \wedge n \ne j : x.n \le y.j \rangle$$

$\equiv$ { split off $j = n$ in the first conjunct and simplify the second one }

---

f Indeed, this has been a hint to our students far more than once

$$\langle \forall i,j : 0 \le i < n \wedge 0 \le j < n \wedge i \ne j : x.i \le y.j \rangle$$
$$\wedge$$
$$\langle \forall i : 0 \le i < n \wedge i \ne n : x.i \le y.n \rangle$$
$$\wedge$$
$$\langle \forall j : 0 \le j < n : x.n \le y.j \rangle$$

$\equiv$ { using loop invariant P1 -below- for the first conjunct, and loop invariant P0 -below-, in particular $1 \le n$, for the two other conjuncts }

$$t \wedge \langle \uparrow i : 0 \le i < n : x.i \rangle \le y.n$$
$$\wedge \quad x.n \le \langle \downarrow j : 0 \le j < n : y.j \rangle \ .$$

The main invariants for our repetition are

P0: $\quad 1 \le n \le N$ $\qquad$ and

P1: $\quad t \equiv \langle \forall i,j : 0 \le i < n \wedge 0 \le j < n \wedge i \ne j : x.i \le y.j \rangle$,

and the above calculation shows that it is beneficial to strengthen them with

P2: $\quad f = \langle \uparrow i : 0 \le i < n : x.i \rangle$ $\qquad$ and

P3: $\quad g = \langle \downarrow j : 0 \le j < n : y.j \rangle$.

The rest of the development is absolutely standard (and not that fascinating).
The program reads

```
  {1 ≤ N}
  n, t, f, g := 1, true, x.0 , y.0
; {inv P0..3}{vf N−n}
  do n ≠ N ∧ t
  →     t, f, g := (t ∧) f≤y.n ∧ x.n≤g
                  , f↑x.n
                  , g↓y.n
      ; n := n+1
  od
```

* * *

Thanks to the ETAC, to Lex in particular, and thanks to Edsger for challenging us, and, of course, to Jayadev for giving us this beautiful little problem.

W.H.J. Feijen,
24 March 1990