The Welfare Crook taken as an exercise in problem decomposition

We consider three ascending sequences $f.\ g.\ h\ .(i: 0 \leq i)$. Given is that these sequences have a value in common; in its most neutral form we can render this as

(0) $\quad (\exists v, i, j, k :: v = f.i \wedge v = g.j \wedge v = h.k)$.

(Dummies range over the obvious domains.) Our problem is to design a program that computes "the least value that the sequences have in common and its first occurrence in each of the sequences".

*

Our purpose is to give a formal derivation of a program meeting the specification. The first thing to do then is to find a formal characterization of what the program has to accomplish. Given (0) serves as a starting point.

It is required that the program computes "the least common value". In order to formalize that aspect we rewrite (0) into

$$(\exists v :: (\exists i, j, k :: v = f.i \wedge v = g.j \wedge v = h.k)),$$

or -- with $c.v$ as a shorthand for the inner quantification -- into

$$(\exists v :: c.v)\ .$$

Now we can characterize that "least value" as

the value U defined by

(U0) $\quad c.U$

(U1) $\quad (\underline{A}\, v:\ c.v :\ v \geq U)$.

(The possibility to define such a least value rests on the well-orderedness of the set of all sequence values.)

Next, it is required that for this U the program computes its "first occurrence" in each of the three sequences. In order to formalize this aspect we first rewrite $c.v$ into a more disentangled form -- using nesting and distribution of $\wedge$ over $\underline{E}$ --

$c.v$: $\quad (\underline{E}i::\ v = f.i) \wedge (\underline{E}j::\ v = g.j) \wedge (\underline{E}k::\ v = h.k)$.

By (U0) we then have

$$(\underline{E}i::\ U = f.i)\ ,$$

and now we can characterize that "first occurrence" in the f-sequence as the value X defined by

(X0) $\quad 0 \leq X$

(X1) $\quad U = f.X$

(X2) $\quad (\underline{A}\, i:\ 0 \leq i \wedge U = f.i :\ i \geq X)$.

(It goes without saying that, by symmetry, we now also have defined Y and Z.)

After thus having decomposed precondition (0), we can formally express the program's

postcondition. It is

R: $x = X \wedge y = Y \wedge z = Z$.

So much for the formal specification of the problem. Now we can start designing a program.

+

Our -- by now standard -- first approximation for program Welfare Crook is

$\{ (X0) \wedge \text{etc.} \}$
$x, y, z := 0, 0, 0$
; do $x < X \rightarrow x := x + 1$
[] $y < Y \rightarrow y := y + 1$
[] $z < Z \rightarrow z := z + 1$
od
$\{ R \}$,

which maintains relation P given by

P: $0 \leq x \wedge x \leq X \wedge$ etc. .

Our -- by now standard -- next step is to strengthen the guards in order to eliminate the occurrences of the unknowns X, Y, Z. To that end we observe

$x < X$
$\Leftarrow$ { f is ascending }
$f.x < f.X$
$=$ { by (X1) and (Y1) }
$f.x < g.Y$
$\Leftarrow$ { transitivity }
$f.x < g.y \wedge g.y \leq g.Y$

$=$ { g is ascending, $y \leq Y$ (by P), hence $g.y \leq g.Y$ }
$f.x < g.y$.

Our program becomes

```
x, y, z := 0, 0, 0
; do  f.x < g.y  →  x := x + 1
  []  g.y < h.z  →  y := y + 1
  []  h.z < f.x  →  z := z + 1
  od .
```

Upon termination it has established

R': $P \wedge f.x = g.y \wedge g.y = h.z \wedge h.z = f.x$.

Our last step is to show that R' is still strong enough to imply target relation R. We observe

$x = X$
$=$ { by P, $x \leq X$ }
$x \geq X$
$\Leftarrow$ { by (X2) with $i := x$ }
$0 \leq x \wedge U = f.x$
$=$ { by P, $0 \leq x$ }
$U = f.x$
$=$ { heading for (U1) }
$f.x \geq U \wedge f.x \leq U$
$=$ { alas (X1) again }
$f.x \geq U \wedge f.x \leq f.X$
$=$ { f is ascending, $x \leq X$ (by P), hence $f.x \leq f.X$ }
$f.x \geq U$
$\Leftarrow$ { by (U1) }

$$
\begin{array}{ll}
 & c.(f.x) \\
= & \quad \{ \text{by definition of } c \text{ with } v := f.x \} \\
\circledast & (\mathbin{\mathbb{E}} i :: f.x = f.i) \wedge (\mathbin{\mathbb{E}} j :: f.x = g.j) \wedge (\mathbin{\mathbb{E}} k :: f.x = h.k) \\
\Leftarrow & \quad \{ \text{instantiation} \} \\
 & f.x = g.y \wedge f.x = h.z \\
= & \quad \{ \text{by } R' \} \\
 & \text{true} .
\end{array}
$$

This completes this derivation. A few remarks, though, are in order.

- The above calculation I find disappointingly long. Here we pay the price for the introduction of parameter U. Of course we could have tried to eliminate it in an earlier stage, but then we would have had to invest formal labour there, without having good reasons for doing so. The question arises whether we could have avoided its introduction in the first place.
- The occurrence of the first conjunct in $\circledast$ makes me suspect that we have named the wrong thing.
- Something nice about the above derivation is that it contains only two formulae that mention all three sequences, viz. c and R', and these "meet" each other precisely once at the very end of the calculation.
- Also nice is that no quantified expressions enter our calculations.
- We should have dealt with the above problem for the slightly more general case of three

ascending chains in a partially order set. We have used the comparability between values of different sequences at only one place (without mentioning it: shame!).

- I expect and I hope that a crispier treatment of the Welfare Crook will emerge (, not with a coarser grain of formality please ; I do not like a 1982 - performance).

Austin,
23 April 1988

W.H.J. Feijen
Department of Computer Sciences,
The University of Texas at Austin,
Austin, TX 78712 - 1188
U.S.A.