Chapter 17 : The Minimal Segment Product.

In which we calculate a beautiful algorithm.

We are given f[0..N) of int, $\{0 \le N\}$ which contains values (not including 0) and we are asked to construct an algorithm to compute the minimal segment product. Our specification is as follows.

$\{f[0..N) \text{ has values }\}$

S

$\{R = \langle \downarrow i,j : 0 \le i \le j \le N : SP.i.j \rangle\}$

Domain modelling.
We begin with the definition of SP

* (0) $SP.i.j \quad = \quad \langle * \ k : i \le k < j : f.k \rangle \qquad , 0 \le i \le j \le N$

From this the following theorems emerge easily. They are stated without proof.

- (1) $SP.i.i \quad = \quad 1 \qquad , 0 \le i \le N$

- (2) $SP.i.(j+1) \quad = \quad SP.i.j * f.j \qquad , 0 \le i \le j < N$

Now let us return to considering the postcondition. Its shape suggests that we name and conquer as follows.

- (3) $C.n \quad = \quad \langle \downarrow i,j : 0 \le i \le j \le n : SP.i.j \rangle \qquad , 0 \le n \le N$

The bounds on the range suggest that instead of considering emptying the range we opt instead to shrink it to one point. Thus we get

- (4) $C.0 \quad = \quad 1$

Next we look at exploiting associativity. We observe

$C.(n+1)$
= $\{(3)\}$
$\langle \downarrow i,j : 0 \le i \le j \le n+1 : SP.i.j \rangle$
= { split off j=n+1 term }
$\langle \downarrow i,j : 0 \le i \le j \le n : SP.i.j \rangle \downarrow \langle \downarrow i : 0 \le i \le n+1 : SP.i.(n+1) \rangle$
= $\{(3)\}$
$C.n \downarrow \langle \downarrow i : 0 \le i \le n+1 : SP.i.(n+1) \rangle$
= { name and conquer see (6) below)

$C.n \downarrow D.(n+1)$

So we have

- (5) $C.(n+1) \quad = \quad C.n \downarrow D.(n+1) \qquad , 0 \leq n < N$

* (6) $D.n \quad = \quad \langle \downarrow i : 0 \leq i \leq n : SP.i.n \rangle, 0 \leq n \leq N$

Now let us explore the properties of D. Appealing to the one point law we have

- (7) $D.0 \quad = \quad 1$

We now appeal to associativity and observe

$$
\begin{array}{ll}
 & D.(n+1) \\
= & \qquad \{(6)\} \\
 & \langle \downarrow i : 0 \leq i \leq n+1 : SP.i.(n+1) \rangle \\
= & \qquad \{\text{split off } i=n+1 \text{ term}\} \\
 & \langle \downarrow i : 0 \leq i \leq n : SP.i.(n+1) \rangle \downarrow SP.(n+1).(n+1) \\
= & \qquad \{(1)\} \\
 & \langle \downarrow i : 0 \leq i \leq n : SP.i.(n+1) \rangle \downarrow 1 \\
= & \qquad \{(2)\} \\
 & \langle \downarrow i : 0 \leq i \leq n : SP.i.n * f.n \rangle \downarrow 1 \\
= & \qquad \{\text{case analysis } 0 < f.n \text{ LAW } (14)\} \\
 & (\langle \downarrow i : 0 \leq i \leq n : SP.i.n \rangle * f.n) \downarrow 1 \\
= & \qquad \{(6)\} \\
 & (D.n * f.n) \downarrow 1
\end{array}
$$

So we have

- (8) $D.(n+1) \quad = \quad (D.n * f.n) \downarrow 1 \quad \Leftarrow \quad 0 \leq f.n \qquad , 0 \leq n < N$

We consider the other case now

$$
\begin{array}{ll}
 & D.(n+1) \\
= & \qquad \{(6)\} \\
 & \langle \downarrow i : 0 \leq i \leq n+1 : SP.i.(n+1) \rangle \\
= & \qquad \{\text{split off } i=n+1 \text{ term}\} \\
 & \langle \downarrow i : 0 \leq i \leq n : SP.i.(n+1) \rangle \downarrow SP.(n+1).(n+1) \\
= & \qquad \{(1)\} \\
 & \langle \downarrow i : 0 \leq i \leq n : SP.i.(n+1) \rangle \downarrow 1 \\
= & \qquad \{(2)\} \\
 & \langle \downarrow i : 0 \leq i \leq n : SP.i.n * f.n \rangle \downarrow 1
\end{array}
$$

= {case analysis f.n < 0 , (15)}

$(\langle \uparrow i : 0 \le i \le n : SP.i.n \rangle * f.n) \downarrow 1$

= {name and conquer see (10 below)}

$(G.n * f.n) \downarrow 1$

Which gives us

- (9) $D.(n+1) = (G.n * f.n) \downarrow 1 \quad \Leftarrow \quad f.n \le 0 \quad , 0 \le n < N$

As we have introduced a new "named" item we explore it

* (10) $G.n = \langle \uparrow i : 0 \le i \le n : SP.i.n \rangle \quad , 0 \le n \le N$

And trivially we deduce

- (11) $G.0 = 1$

We explore appealing to associativity by observing

$G.(n+1)$

= {(10)}

$\langle \uparrow i : 0 \le i \le n+1 : SP.i.(n+1)\rangle$

= {split off i = n+1 term}

$\langle \uparrow i : 0 \le i \le n : SP.i.(n+1)\rangle \uparrow SP.(n+1).(n+1)$

= {(1)}

$\langle \uparrow i : 0 \le i \le n : SP.i.(n+1)\rangle \uparrow 1$

= {(2)}

$\langle \uparrow i : 0 \le i \le n : SP.i.n * f.n\rangle \uparrow 1$

= {case analysis 0 < f.n , (16)}

$(\langle \uparrow i : 0 \le i \le n : SP.i.n\rangle * f.n) \uparrow 1$

= {(10)}

$(G.n * f.n) \uparrow 1$

So we have

- (12) $G.(n+1) = (G.n * f.n) \uparrow 1 \quad \Leftarrow \quad 0 \le f.n \quad , 0 \le n < N$

And we explore the other case

$G.(n+1)$

= {(10)}

$\langle \uparrow i : 0 \le i \le n+1 : SP.i.(n+1)\rangle$

= {split off i = n+1 term}

$\langle \uparrow i : 0 \le i \le n : SP.i.(n+1) \rangle \uparrow SP.(n+1).(n+1)$

= {(1)}

$\langle \uparrow i : 0 \le i \le n : SP.i.(n+1) \rangle \uparrow 1$

= {(2)}

$\langle \uparrow i : 0 \le i \le n : SP.i.n * f.n \rangle \uparrow 1$

= {case analysis f.n < 0 , (17)}

$(\langle \downarrow i : 0 \le i \le n : SP.i.n \rangle * f.n) \uparrow 1$

= {(6)}

$(D.n * f.n) \uparrow 1$

Which can be expressed as

- (13) $G.(n+1) = (D.n * f.n) \uparrow 1 \quad \Leftarrow \quad f.n \le 0 \quad , 0 \le n < N$

And this completes our mathematical model. We now return to our original programming task.

Rewrite the postcondition using the model.

POST : R = C.N

Using strengthening we get

POST' : $r = C.n \wedge n=N$

Invariants.

P0 : $r = C.n \wedge d = D.n \wedge g = G.n$ [1]

P1 : $0 \le n \le N$

We note that

$P0 \wedge P1 \wedge n=N \quad \Rightarrow \quad POST$

Guard.

$n \neq N$

---

[1] Why not just use R = C.n? Well in developing our model we discovered that some of the laws about C refer to D and also some of the laws of D refer to G. So it seems reasonable to “carry” the values of D.n and G.n with us.

Establishing the invariants.

Looking at our model we note that laws (4), (7) and (11) suggest the following assignment

$$n, R, d, g := 0, 1, 1, 1$$

Variant function.

We choose N-n as the variant and it is a standard argument to show that this can be decreased by the assignment n := n+1 while maintaining P1.

Calculating the loop body.

Our calculation proceeds as follows by case analysis

$$
\begin{array}{ll}
 & (n, R, d, g := n+1, E, E', E'').P0 \\
= & \quad \{\text{textual substitution}\} \\
 & E = C.(n+1) \wedge E' = D.(n+1) \wedge E'' = G.(n+1) \\
= & \quad \{(5)\} \\
 & E = C.n \downarrow D.(n+1) \wedge E' = D.(n+1) \wedge E'' = G.(n+1) \\
= & \quad \{\text{case analysis } 0 < f.n\text{, laws (8), (12)}\} \\
 & E = C.n \downarrow (D.n * f.n) \downarrow 1 \wedge E' = (D.n * f.n) \downarrow 1 \wedge E'' = (G.n * f.n) \uparrow 1 \\
= & \quad \{P0\} \\
 & E = R \downarrow (d * f.n) \downarrow 1 \wedge E' = (d * f.n) \downarrow 1 \wedge E'' = (g * f.n) \uparrow 1
\end{array}
$$

Which gives us

If $0 \le f.n \rightarrow n, R, d, g := n+1, R \downarrow (d * f.n) \downarrow 1, (d * f.n) \downarrow 1, (g * f.n) \uparrow 1$

We now consider the other case

$$
\begin{array}{ll}
 & (n, R, d, g := n+1, E, E', E'').P0 \\
= & \quad \{\text{textual substitution}\} \\
 & E = C.(n+1) \wedge E' = D.(n+1) \wedge E'' = G.(n+1) \\
= & \quad \{(5)\} \\
 & E = C.n \downarrow D.(n+1) \wedge E' = D.(n+1) \wedge E'' = G.(n+1) \\
= & \quad \{\text{case analysis } f.n < 0\text{, laws (9), (12)}\} \\
 & E = C.n \downarrow (G.n * f.n) \downarrow 1 \wedge E' = (G.n * f.n) \downarrow 1 \wedge E'' = (D.n * f.n) \uparrow 1 \\
= & \quad \{p0\} \\
 & E = R \downarrow (g * f.n) \downarrow 1 \wedge E' = (g * f.n) \downarrow 1 \wedge E'' = (d * f.n) \uparrow 1
\end{array}
$$

which gives us

If f.n ≤ 0 → n, R, d, g := n+1, R ↓ (g * f.n) ↓ 1, (g * f.n) ↓ 1, (d * f.n) ↑ 1

Finished program.

```
n, r, d, g := 0,1,1,1
;do n≠N →      {P0 ∧ P1 ∧ n≠N }

               If 0 < f.n → n, R, d, g :=     n+1,
                                              R ↓ (d * f.n) ↓ 1,
                                              (d * f.n) ↓ 1,
                                              (g * f.n) ↑ 1

               [] f.n < 0 → n, R, d, g :=     n+1,
                                              R ↓ (g * f.n) ↓ 1,
                                              (g * f.n) ↓ 1,
                                              (d * f.n) ↑ 1
               fi

               {P0 ∧ P1}
od
{P0 ∧ P1 ∧ n=N}
```

Appendix.

Here is our full mathematical model. In our modelling we appealed to a number of distribution laws which we list below (14) – (17). Note that these apply as long as the range of the quantification is non-empty.

* (0) SP.i.j = $\langle * \ k : i \le k < j : f.k \rangle$, $0 \le i \le j \le N$

\- (1) SP.i.i = 1 , $0 \le i \le N$

\- (2) SP.i.(j+1) = SP.i.j * f.j , $0 \le i \le j < N$

\- (3) C.n = $\langle \downarrow \ i,j : 0 \le i \le j \le n : SP.i.j \rangle$, $0 \le n \le N$

- (4) $C.0 = 1$

- (5) $C.(n+1) = C.n \downarrow D.(n+1)$, $0 \le n < N$

* (6) $D.n = \langle \downarrow i : 0 \le i \le n : SP.i.n \rangle$, $0 \le n \le N$

- (7) $D.0 = 1$

- (8) $D.(n+1) = (D.n * f.n) \downarrow 1 \quad \Leftarrow \quad 0 \le f.n$, $0 \le n < N$

- (9) $D.(n+1) = (G.n * f.n) \downarrow 1 \quad \Leftarrow \quad f.n \le 0$, $0 \le n < N$

* (10) $G.n = \langle \uparrow i : 0 \le i \le n : SP.i.n \rangle$, $0 \le n \le N$

- (11) $G.0 = 1$

- (12) $G.(n+1) = (G.n * f.n) \uparrow 1 \quad \Leftarrow \quad 0 \le f.n$, $0 \le n < N$

- (13) $G.(n+1) = (D.n * f.n) \uparrow 1 \quad \Leftarrow \quad f.n \le 0$, $0 \le n < N$

* (14) $X * \langle \downarrow i : 0 \le i \le n : f.i \rangle = \langle \downarrow i : 0 \le i \le n : X * f.i \rangle \Leftarrow 0 \le X \wedge 0 \le n$

* (15) $X * \langle \uparrow i : 0 \le i \le n : f.i \rangle = \langle \downarrow i : 0 \le i \le n : X * f.i \rangle \Leftarrow X \le 0 \wedge 0 \le n$

* (16) $X * \langle \uparrow i : 0 \le i \le n : f.i \rangle = \langle \uparrow i : 0 \le i \le n : X * f.i \rangle \Leftarrow 0 \le X \wedge 0 \le n$

* (17) $X * \langle \uparrow i : 0 \le i \le n : f.i \rangle = \langle \downarrow i : 0 \le i \le n : X * f.i \rangle \Leftarrow X \le 0 \wedge 0 \le n$