Chapter 8: The Partitioned Reduction Theorem I.

In which we introduce another generic problem.

We are given an array f[0..N) of int which contains values and we are asked to construct a program to count the number of even values in f. We begin by specifying the problem.

$\{$ f[0..N) contains values $\}$

S

$\{r = \langle + j : 0 \le j < N : g.(f.j) \rangle\}$

where

g.x	=	1	$\Leftarrow$	even.x
g.x	=	0	$\Leftarrow$	odd.x

Postcondition.

$$\text{Post} : r = \langle + j : 0 \le j < N : g.(f.j) \rangle$$

Strengthen postcondition.

$$\text{Post'} : r = \langle + j : 0 \le j < n : g.(f.j) \rangle \wedge n = N$$

Domain modelling.

Inspired by the shape of our postcondition, we now proceed to develop a little mathematical model of our domain. We begin with a single postulate.

$$* (0) \quad C.n \quad = \quad \langle + j : 0 \le j < n : g.(f.j) \rangle \qquad , 0 \le n \le N$$

The function g is defined as follows:

$$* (1) \quad g.x \quad = \quad 1 \qquad \Leftarrow \qquad even.x$$

$$* (2) \quad g.x \quad = \quad 0 \qquad \Leftarrow \qquad odd.x$$

We now explore some theorems.

Consider

$$C.0$$
$$=\qquad \{(0)\ \text{in model}\ \}$$
$$\langle + j : 0 \le j < 0 : g.(f.j)\rangle$$
$$=\qquad \{\ \text{empty range}\ \}$$
$$Id+$$

Which gives us

- (3) $C.0 \quad = \quad Id+$

Consider

$$C.(n+1)$$
$$=\qquad \{(0)\ \text{in model}\}$$
$$\langle + j : 0 \le j < n+1 : g.(f.j)\rangle$$
$$=\qquad \{\text{split off}\ j = n\ \text{term}\ \}$$
$$\langle + j : 0 \le j < n : g.(f.j)\rangle + g.(f.n)$$
$$=\qquad \{\ \text{case even}.(f.n),\ (1)\ \text{in model}\ \}$$
$$\langle + j : 0 \le j < n : g.(f.j)\rangle + 1$$

Which gives us

- (4) $C.(n+1) = \quad C.n + 1 \qquad \Leftarrow \qquad \text{even}.(f.n) \qquad , 0 \le n < N$

Now consider the other case.

$$C.(n+1)$$
$$=\qquad \{(0)\ \text{in model}\}$$
$$\langle + j : 0 \le j < n+1 : g.(f.j)\rangle$$
$$=\qquad \{\text{split off}\ j = n\ \text{term}\ \}$$
$$\langle + j : 0 \le j < n : g.(f.j)\rangle + g.(f.n)$$
$$=\qquad \{\ \text{case odd}.(f.n),\ (2)\ \text{in model}\ \}$$
$$\langle + j : 0 \le j < n : g.(f.j)\rangle + 0$$

Which gives us

- (5) C.(n+1) = C.n + 0 ⟸ odd.(f.n) , 0 ≤ n < N

This completes our model.

Rewrite postcondition in terms of model.

Invariants.

   P0 : r = C.n

   P1 : 0 ≤ n ≤ N

Termination.

We observe that

   P0 ∧ P1 ∧ n=N ⟹ Post

Establishing the invariants.

To establish P0 we need to bind r to the value of C.n, for some n. Theorem (1) in our model gives us the value of C.n when n=0. So the following assignment establishes P0 and also P1.

   n, r := 0, Id+

Guard.

   n ≠ N

Variant.

   N-n

Loop body.

       (n, r := n+1, E).P0

=   { textual substitution }

       E = C.(n+1)

=   {Case analysis, even.(f.n) , P1 and n ≠ N allow us to appeal to (4)}

       E = C.n + 1

=   {P0 }

       E = r + 1

Giving us the program fragment

       [] even.(f.n)   →   n, r := n+1, r+1

We now look at the other case.

       (n, r := n+1, E).P0

=   { textual substitution }

       E = C.(n+1)

=   {Case analysis, odd.(f.n) , P1 and n ≠ N allow us to appeal to (5)}

       E = C.n + 0

=   {P0 }

       E = r + 0

Giving us the program fragment

       [] even.(f.n)   →   n, r := n+1, r+0

As (even.x ∨ odd.x) ≡ true we have covered all possibilities so we can now write the finished loop program.

Finished program.

      n, r := 0, Id+

      ;do n ≠ N   →

                        If even.(f.n)   → n, r := n+1, r+1

                        [] odd.(f.n)   → n, r := n+1, r+0

                        Fi

      od

      {P0 ∧ P1 ∧ n = N }

Exercises.

Given an array f[0..N) of int, where {0 ≤ N}, calculate programs to do the following.

- sum the odd values in f
- determine the largest even value in f.
- develop the Generic solution to this class of problem.