Chapter 14§ : Fusc.

In which we apply the techniques we learned in Chapter 13.

Suppose we are given the fusc function defined as follows.

fusc : natural → natural

* (0) fusc.0 = 0

* (1) fusc.1 = 1

* (2) fusc.(2*n) = fusc.n , 0 < n

* (3) fusc.(2*n+1) = fusc.n + fusc.(n+1) , 0 < n

We are given a natural number N and asked to write a program to compute fusc.N

Choosing invariants.

We propose as invariants

$P0 : \alpha * fusc.n + \beta * fusc.(n+1) = fusc.N$
$P1 : 0 \leq n \leq N$

Establishing the Invariants.

The following assignment establishes the invariants

$n, \alpha, \beta := N, 1, 0$

Termination.

We note the following.

$P0 \wedge P1 \wedge n=0$
= {definitions of P0, P1}
$\alpha * fusc.n + \beta * fusc.(n+1) = fusc.N \ \wedge \ 0 \leq n \leq N \ \wedge n=0$
⇒ {leibniz}
$\alpha * fusc.0 + \beta * fusc.1 = fusc.N \ \wedge$ true
= {predicate calculus}
$\alpha * fusc.0 + \beta * fusc.1 = fusc.N$

= {defn. fusc}

$\alpha * 0 + \beta * 1 = \text{fusc.N}$

= {arithmetic}

$\beta = \text{fusc.N}$

Guard.

$n \neq 0$

Variant.

n

Calculate the loop body.

P0

= {definition P0}

$\alpha*\text{fusc.n} + \beta*\text{fusc.}(n+1) = \text{fusc.N}$

= {case analysis, even.n i.e. n = 2*p}

$\alpha*\text{fusc.}(2*p) + \beta*\text{fusc.}(2*p+1) = \text{fusc.N}$

= {definition fusc (2) (3)}

$\alpha * \text{fusc.p} + \beta*(\text{fusc.p} + \text{fusc.}(p+1)) = \text{fusc.N}$

= {*/+}

$\alpha * \text{fusc.p} + \beta*\text{fusc.p} + \beta*\text{fusc.}(p+1) = \text{fusc.N}$

= {gather like terms together}

$(\alpha+\beta)*\text{fusc.p} + \beta*\text{fusc.}(p+1) = \text{fusc.N}$

= {WP}

$(n, \alpha, \beta := n \text{ div } 2, \alpha + \beta, \beta).\text{P0}$

Giving us the program fragment

if even.n → $n, \alpha, \beta := n \text{ div } 2, \alpha + \beta, \beta$

Let us analyse the other case

P0

= {definition P0}

$\alpha*\text{fusc.n} + \beta*\text{fusc.}(n+1) = \text{fusc.N}$

= {case analysis, odd.n i.e. n = 2*p+1}

$\alpha*\text{fusc.}(2*p+1) + \beta*\text{fusc.}(2*p+2) = \text{fusc.N}$

= {definition fusc (2) (3)}

$\alpha*(\text{fusc.p} + \text{fusc.}(p+1)) + \beta*\text{fusc.}(p+1) = \text{fusc.N}$

= {*/+}

$\alpha*\text{fusc.p} + \alpha*\text{fusc.}(p+1) + \beta*\text{fusc.}(p+1) = \text{fusc.N}$

= {gather like terms together}

α*fusc.p + ($\alpha+\beta$)*fusc.(p+1) = fusc.N
= {WP}
(n, α, β := (n-1) div 2, α , $\alpha + \beta$).P0

Which gives us the program fragment

```
if odd.n → n, $\alpha$, $\beta$ := (n-1) div 2, $\alpha$ , $\alpha + \beta$
```

Finished program.

And so the entire program is as follows.

```
n, $\alpha$, $\beta$ := N, 1, 0 {P0 $\wedge$ P1}
;do n $\neq$ 0  →    {P0 $\wedge$ P1 $\wedge$ n $\neq$ 0}

                    if even.n → n, $\alpha$, $\beta$ := n div 2, $\alpha + \beta$, $\beta$
                    [] odd.n → n, $\alpha$, $\beta$ := (n-1) div 2, $\alpha$ , $\alpha + \beta$
                    fi

                    {P0 $\wedge$ P1}
od
{$\beta$ = fusc.N}
```

We note that this solution has temporal complexity O(log(N))