```
#|
```

```
|#

;               Case study: Switch Statement
```

EVENT: Start with the library "mc20-2" using the compiled version.

```
#|

The purpose of this trivial C function here is to study the switch construct
in C.

int foo(int n)
{
  int i;

  switch(n) {
  case 0: i = 0; break;
  case 1: i = 1; break;
  case 2: i = 4; break;
  case 3: i = 9; break;
  case 4: i = 16; break;
  default: i = n; break;
  };
  return i;
}

Here is the MC68020 assembly code of the above function.  The code is
generated by gcc with optimization option.

0x23b2 <foo>:      linkw a6,#0
0x23b6 <foo+4>:    movel a6@(8),d0
0x23ba <foo+8>:    movel #4,d1
0x23bc <foo+10>:   cmpl d1,d0
0x23be <foo+12>:   bhi 0x23e4 <foo+50>
0x23c0 <foo+14>:   movew 0x23c8[d0.l*2],d1
0x23c4 <foo+18>:   jmp 0x23c8[d1.w]
0x23c8 <foo+22>:   orb #14,a2
0x23cc <foo+26>:   orb #22,a2@
0x23d0 <foo+30>:   orb #-128,a2@+
0x23d4 <foo+34>:   bra 0x23e4 <foo+50>
0x23d6 <foo+36>:   movel #1,d0
0x23d8 <foo+38>:   bra 0x23e4 <foo+50>
0x23da <foo+40>:   movel #4,d0
0x23dc <foo+42>:   bra 0x23e4 <foo+50>
0x23de <foo+44>:   movel #9,d0
0x23e0 <foo+46>:   bra 0x23e4 <foo+50>
0x23e2 <foo+48>:   movel #16,d0
0x23e4 <foo+50>:   unlk a6
0x23e6 <foo+52>:   rts

The machine code of the above program is:

<foo>:       0x4e56  0x0000  0x202e  0x0008  0x7204  0xb081  0x6224  0x323b
<foo+16>:    0x0a06  0x4efb  0x1002  0x000a  0x000e  0x0012  0x0016  0x001a
<foo+32>:    0x4280  0x600e  0x7001  0x600a  0x7004  0x6006  0x7009  0x6002
<foo+48>:    0x7010  0x4e5e  0x4e75

'(78      86      0       0       32      46      0       8
  114     4       176     129     98      36      50      59
  10      6       78      251     16      2       0       10
  0       14      0       18      0       22      0       26
  66      128     96      14      112     1       96      10
  112     4       96      6       112     9       96      2
  112     16      78      94      78      117)
|#
```

```
; in the logic, the above program is specified as (foo-code).
```

Definition:
FOO-CODE
= `'(78 86 0 0 32 46 0 8 114 4 176 129 98 36 50 59 10 6 78 251 16 2 0 10 0 14 0 18 0 22 0 26 66 128 96 14 112 1 96 10 112 4 96 6 112 9 96 2 112 16 78 94 78 117)`

Definition:
foo (n)
= **if** between-ileq (0, n, 4) **then** $n * n$
else n **endif**

Definition:
foo-t (n)
= **if** $(n = 0) \lor (n = 1) \lor (n = 2) \lor (n = 3)$ **then** 11
elseif $n = 4$ **then** 10
else 7 **endif**

Definition:
foo-statep $(s,\ n)$
= ((mc-status (s) = `'running`)
∧ evenp (mc-pc (s))
∧ rom-addrp (mc-pc (s), mc-mem (s), 54)
∧ mcode-addrp (mc-pc (s), mc-mem (s), FOO-CODE)
∧ ram-addrp (sub (32, 4, read-sp (s)), mc-mem (s), 12)
∧ disjoint (mc-pc (s), 54, sub (32, 4, read-sp (s)), 12)
∧ (n = iread-mem (add (32, read-sp (s), 4), mc-mem (s), 4)))

Definition:
foo-snp $(s,\ sn,\ n,\ oplen,\ rn,\ x,\ k)$
= ((mc-status (sn) = `'running`)
∧ (mc-pc (sn) = rts-addr (s))
∧ (iread-dn (32, 0, sn) = foo (n))
∧ (read-rn (32, 14, mc-rfile (sn))
= read-rn (32, 14, mc-rfile (s)))
∧ (read-rn (32, 15, mc-rfile (sn)) = add (32, read-an (32, 7, s), 4))
∧ (read-rn ($oplen$, rn, mc-rfile (sn))
= read-rn ($oplen$, rn, mc-rfile (s)))
∧ (read-mem (x, mc-mem (sn), k) = read-mem (x, mc-mem (s), k)))

Theorem: foo-s-sn
(foo-statep $(s,\ n)$

$$
\begin{array}{ll}
\wedge & \text{d2-7a2-5p}\,(rn) \\
\wedge & \text{disjoint}\,(x,\ k,\ \text{sub}\,(\mathtt{32},\ \mathtt{4},\ \text{read-sp}\,(s)),\ \mathtt{12})) \\
\rightarrow & \text{foo-snp}\,(s,\ \text{stepn}\,(s,\ \text{foo-t}\,(n)),\ n,\ oplen,\ rn,\ x,\ k)
\end{array}
$$

Theorem: foo-correctness

$$
\begin{array}{l}
\textbf{let}\ sn\ \ \textbf{be}\ \ \text{stepn}\,(s,\ \text{foo-t}\,(n)) \\
\textbf{in} \\
\text{foo-statep}\,(s,\ n) \\
\rightarrow\quad ((\text{mc-status}\,(sn) = \mathtt{'running}) \\
\qquad\quad \wedge\quad (\text{mc-pc}\,(sn) = \text{rts-addr}\,(s)) \\
\qquad\quad \wedge\quad (\text{read-rn}\,(\mathtt{32},\ \mathtt{14},\ \text{mc-rfile}\,(sn)) \\
\qquad\qquad\quad =\quad \text{read-rn}\,(\mathtt{32},\ \mathtt{14},\ \text{mc-rfile}\,(s))) \\
\qquad\quad \wedge\quad (\text{read-rn}\,(\mathtt{32},\ \mathtt{15},\ \text{mc-rfile}\,(sn)) \\
\qquad\qquad\quad =\quad \text{add}\,(\mathtt{32},\ \text{read-an}\,(\mathtt{32},\ \mathtt{7},\ s),\ \mathtt{4})) \\
\qquad\quad \wedge\quad (\text{d2-7a2-5p}\,(rn) \\
\qquad\qquad\quad \rightarrow\quad (\text{read-rn}\,(oplen,\ rn,\ \text{mc-rfile}\,(sn)) \\
\qquad\qquad\qquad\quad =\quad \text{read-rn}\,(oplen,\ rn,\ \text{mc-rfile}\,(s)))) \\
\qquad\quad \wedge\quad (\text{disjoint}\,(x,\ k,\ \text{sub}\,(\mathtt{32},\ \mathtt{4},\ \text{read-sp}\,(s)),\ \mathtt{12}) \\
\qquad\qquad\quad \rightarrow\quad (\text{read-mem}\,(x,\ \text{mc-mem}\,(sn),\ k) \\
\qquad\qquad\qquad\quad =\quad \text{read-mem}\,(x,\ \text{mc-mem}\,(s),\ k))) \\
\qquad\quad \wedge\quad (\text{iread-dn}\,(\mathtt{32},\ \mathtt{0},\ sn) = \text{foo}\,(n)))\ \textbf{endlet}
\end{array}
$$

Index