Prove by strong mathematical induction on $i$:

$\forall i, C, s, s'.\ \langle C, s\rangle \Downarrow_i s' \implies \langle f(C), s\rangle \Downarrow s'$.

※ Lemma 1: $\forall B, s:\ \langle B, s\rangle \Downarrow_b \text{false} \implies \langle \neg B, s\rangle \Downarrow_b \text{true}$.

$\boxed{\text{Base Case}}$: $i=0$.

Only 3 cases: ① $\langle \text{skip}, s\rangle \Downarrow_0 s$ ($C=\text{skip}$), ② $\langle x:=E, s\rangle \Downarrow_0 s'$ ($C=x:=E$), ③ $\langle \text{while } B \text{ do } C_1, s\rangle \Downarrow_0 s$ ($C=\text{while } B \text{ do } C_1$)

from pattern matching on rules of While language.

① To show: $\langle \text{skip}, s\rangle \Downarrow_0 s \implies \langle f(\text{skip}), s\rangle \Downarrow s$.

(a) $f(\text{skip}) = \text{skip}$. by def of $f()$.

(b) $\langle f(\text{skip}), s\rangle \Downarrow s$. by skip rule in NONDET, (a).

② To show $\langle x:=E, s\rangle \Downarrow_0 s' \implies \langle f(x:=E), s\rangle \Downarrow s'$

(a) $f(x:=E) = x:=E$. by def of $f()$

(b) $\langle E, s\rangle \Downarrow_e n$.

(c) $s[x \to n] = s'$

(b), (c): by inversion on Asgn in WHILE

(d) $\langle f(x:=E), s\rangle \Downarrow s'$ by rule of Asgn in NONDET, (b), (c).

③ To show $\langle \text{while } B \text{ do } C_1, s\rangle \Downarrow_0 s \implies \langle f(\text{while } B \text{ do } C_1), s\rangle \Downarrow s$.

(a) $f(\text{while } B \text{ do } C_1) = \text{loop}(\text{assume } B;\ f(C_1));\ \text{assume } \neg B$ by def of $f()$

(b) $\langle B, s\rangle \Downarrow_b \text{false}$ by inversion on while-false in WHILE.

here there's no derivation rule for assume $B$ by (b) and def of NONDET language, we can only apply the second rule for loop.

(c) $\langle \text{loop}(\text{assume } B;\ f(C_1)), s\rangle \Downarrow s$. (reason above)

(d) $\langle \neg B, s\rangle \Downarrow_b$ true by (b) and Lemma 4.

(e) (assume $\neg B$, s) $\Downarrow s$ by (d) and rule of Assume in NONDET

(f) $\langle$loop(assume B; f(c1)) ; assume $\neg B, s\rangle \Downarrow s$ S

by (c), (e) and rule for SEQ in NONDET.

(g) $\langle$f(while B do c1), s$\rangle \Downarrow$ s. by (f) and (a)

## Inductive Step:

IH: $\forall n(\text{s.t. } n<k+1), s, s'$ $\langle c, s\rangle \Downarrow_n s' \Rightarrow \langle f(c), s\rangle \Downarrow s'$

To show: $\forall k+1, s, s'$ $\langle c, s\rangle \Downarrow_{k+1} s' \Rightarrow \langle f(c), s\rangle \Downarrow s'$

① C= C1;C2

To show: $\langle c_1;c_2, s\rangle \Downarrow_{k+1} s' \Rightarrow \langle f(c_1;c_2), s\rangle \Downarrow s'$

(a) : $\langle c_1, s\rangle \Downarrow_i s''$

(b) : $\langle c_2, s''\rangle \Downarrow_j s'$

(c) $k = \max(i, j)$

(a)–(c): by inversion on SEQ rule in WHILE

(d) : $\langle c_1, s\rangle \Downarrow_i s'' \Rightarrow \langle f(c_1), s\rangle \Downarrow s''$

(e) : $\langle c_2, s'\rangle \Downarrow_j s' \Rightarrow \langle f(c_2), s'\rangle \Downarrow s'$

(d)–(e): by (a), (b) since we can pick s, s' arbitrarily from IH. and $i, j < k$ (from (c))

(f) : $f(c_1;c_2) = f(c_1); f(c_2)$ by def of f()

(g) $\langle f(c_1); f(c_2), s\rangle \Downarrow s'$ by (a), (b), (d), (e) and rule for SEQ in NONDET.

(h) $\langle f(c_1;c_2), s\rangle \Downarrow s'$ by (d), (c)

② C= while B do C1.

(a) f(while B do c1) = loop (assume B; f(c1)); assume $\neg B$.

(b) $\langle B, s\rangle \Downarrow_b$ true

(c) $\langle c_1, s\rangle \Downarrow_i s''$

(d) $\langle \text{while } B \text{ do } C_1, s'' \rangle \Downarrow_j s'$

(e) $k = \max(i, j)$

(f) $\langle f(\text{while } B \text{ do } C_1), s'' \rangle \Downarrow s'$ by IH, (d), (e). pick s, s' arbitrary

(g) $\langle f(C_1), s \rangle \Downarrow s''$ by IH, (c), (e) pick s, s' arbitrary.

(i) $\langle \text{assume } B, s \rangle \Downarrow s$ by (b) and rule for Assume in NONDET.

(j) $\langle \text{assume } B; f(C_1), s \rangle \Downarrow s''$ by (g), (i) and rule for SEQ in NONDET.

(k) $\langle \text{loop}(\text{assume } B; f(C_1)); \text{assume } \neg B, s'' \rangle \Downarrow s'$ by def of f() and (f).

(l) $\langle \text{loop}(\text{assume } B; f(C_1)), s'' \rangle \Downarrow s'$ by (k), (b), def of Assume in NONDET and inversion on SEQ in NONDET

(m) $\langle \text{loop}(\text{assume } B; f(C_1)), s \rangle \Downarrow s'$ by (g), (l) and def of while in NONDET.

(n) $\langle \text{assum } \neg B, s' \rangle \Downarrow s'$

※ Here B must eval to false in order to escape from while do based on definition, so we have $\langle \neg B, s' \rangle \Downarrow \text{true}$, n holds

(o) $\langle \text{loop}(\text{assume } B; f(C_1)); \text{assum } \neg B, s \rangle \Downarrow s'$

by (m), (n) and rule for SEQ in NONDET.

(p) $\langle f(\text{while } B \text{ do } C_1), s \rangle \Downarrow s'$

③ C = if B then C1 then C2.

To show $\langle \text{if } B \text{ then } C_1 \text{ else } C_2, s \rangle \Downarrow_{k+1} s'$

$\Rightarrow \langle C(\text{if } B \text{ then } C_1 \text{ else } C_2, s) \rangle \Downarrow s'$

③.1 (a) $\langle B, s \rangle \Downarrow_s \text{true}$

(b) $\langle C_1, s \rangle \Downarrow_k s'$

} by inversion on IF-TRUE in WHILE.

(c) $\langle f(c_1), s\rangle \Downarrow s'$ from IH and (b)

(d) $f(c) = or((assume\ B; f(c_1)), (assume\ \neg B; f(c_2)))$.

(e) $\langle assume\ B, s\rangle \Downarrow s$ from (a) and def of Assume in NONDET.

R here cannot further eval to RHS since no rule for false assumption

(f) $\langle assume\ B; f(c_1), s\rangle \Downarrow s'$ by (e), (c) and def of SEQ in NONDET.

(g) $or((assume\ B; f(c_1)), (assume\ \neg B; f(c_2))) \Downarrow s'$

by (f) and first OR rule in NONDET, and reason R above

(h) $\langle f(c), s\rangle \Downarrow s'$ by (g) and (d)

(Similar for true case)

B.2 (a) $\langle B, s\rangle \Downarrow_s false$

(b) $\langle c_2, s\rangle \Downarrow_k s'$

by inversion on IF-FALSE in WHILE.

(c) $\langle f(c_2), s\rangle \Downarrow s'$ from IH and (b)

(d) $f(c) = or((assume\ B; f(c_1)), (assume\ \neg B; f(c_2)))$.

(e) $\langle \neg B, s\rangle \Downarrow true$ by (a) and Lemma 1

(f) $\langle assume\ \neg B, s\rangle \Downarrow s$ from (a) (e) and def of Assume in NONDET.

R' here cannot further eval to LHS since no rule for false assumption

(g) $\langle assume\ \neg B; f(c_2), s\rangle \Downarrow s'$ by (f), (c) and def of SEQ in NONDET.

(h) $or((assume\ B; f(c_1)), (assume\ \neg B; f(c_2))) \Downarrow s'$

by (g) and second OR rule in NONDET, and reason R' above

(i) $\langle f(c), s\rangle \Downarrow s'$ by (h) and (d)