

14F-1 Bookkeeping

- 0 pts Correct

Peer Review ID: 70947656 — enter this when you fill out your peer evaluation via gradescope

4F-2.

The rule has a bug, because the value of x is changed permanently. The value of x should be recovered after c . To fix this,

I use equivalent statements that let $x = e$ in c :

$tmp := x; x := e; c; x := tmp;$

$\forall c \text{ (let } x := e \text{ in } c, B) = \forall c \text{ (} tmp := x; x := e; c; x := tmp, B)$

$= [x / tmp] \forall c \text{ (} x := e; c; x := tmp, B)$

$= \{x / tmp\} \{e / x\} \forall c \text{ (} c; x := tmp, B)$

$= \{x / tmp\} \{e / x\} \forall c \text{ (} c; B)$

4F-3

1. a command c : let $x = 0$ in $y = x$

2. a post-condition B : $x = 0$

3. a state σ such that $\sigma(x) = 1$

4. $G = \forall c \text{ (} c, B)$

5. $\langle c, \sigma \rangle \not\models G$

$\sigma = \{x=1\}$
 $\sigma' = \{x=0, y=0\}$

6. $\sigma' \neq \sigma$

B does not hold in σ'

2 4F-2 VCGen for Let

- 0 pts Correct

4F-2.

The rule has a bug, because the value of x is changed permanently. The value of x should be recovered after c .
To fix this,

I use equivalent statements that let $x = e$ in c :

tmp := x ; $x := e$; c ; $x := \text{tmp}$;

$\forall c$ (let $x = e$ in c, B) = $\forall c$ (tmp := x ; $x := e$; c ; $x := \text{tmp}$, B)

= [x / tmp] $\forall c$ ($x := e$; c ; $x := \text{tmp}$, B)

= { x / tmp } [e / x] $\forall c$ (c ; $x := \text{tmp}$, B)

= { x / tmp } [e / x] $\forall c$ (c ; temp / B)

4F-3

1. a command c : let $x = 0$ in $y = x$

2. a post-condition B : $x = 0$

3. a state σ such that $\sigma(x) = 1$

4. $G = \forall c (c, B)$

5. $\langle c, \sigma \rangle \not\models G$

$\sigma = \{ x = 1 \}$
 $\sigma' = \{ x = 0, y = 0 \}$

6. $\sigma' \neq \sigma$

B does not hold in σ'

3 4F-3 VCGen Mistakes

- 0 pts Correct

44-4

we know from lecture:

$$\vdash \{A\} c_1 \{C\} \quad \vdash \{C\} c_2 \{B\}$$

$$\vdash \{A\} c_1, c_2 \{B\}$$

$$\vdash \{A \wedge b\} c \{A\}$$

$$\vdash \{A\} \text{ while } b \text{ do } c \{A \wedge \neg b\}$$

because $\text{do } c \text{ while } b$ is equivalent to

$c; \text{ while } b \text{ do } c$

by applying the two rules:

$$\vdash \{A\} c \{B\} \quad \vdash \{B\} \text{ while } B \text{ do } c \{C\}$$

$$\vdash \{A\} \text{ do } c \text{ while } b \{C\}$$

$$\vdash \{A\} c \{B\} \quad \vdash \{B\} \text{ while } b \text{ do } c \{B \wedge \neg b\}$$

$$\vdash \{A\} \text{ do } c \text{ while } b \{B \wedge \neg b\}$$

4 4F-4 Axiomatic Do-While

- 0 pts Correct