《软件分析与验证》第六次书面作业讲解

朱倖民

清华大学

2020 年 6 月

Contents

Contents

1-1
参考解答

Question

Which of the following is *not* a loop invariant for the following IMP loop?

$$\textbf{while } Y > 0 \textbf{ do } Y := Y - 1; X := X + 1 \textbf{ end}$$

(A) $X > 10$.
(B) $Y > 10$.
(C) $X + Y = Z$.
(D) $Z + Y < X$.

(B)。循环体执行前若 $Y = 11$，则执行一次后 $Y = 10$，不再满足 $Y > 10$。不变式定义：使得 $\{b \wedge I\}\ c\ \{I\}$（While 的 Hoare rule 的前提）成立的 I。

Question

Which of the following is *false* about the IMP program shown below?

$X := 1;$
while $X > 0$ **do**
  if $N \leq 100$ **then**
    $N := N + 11;$
    $X := X + 1$
  else
    $N := N - 10;$
    $X := X - 1$
  fi end

(A) $X = 0$ is a post condition.

(B) $X \geq 0$ is a loop invariant (for the while-loop).

(C) $N \leq 111$ is a loop invariant (for the while-loop).

(D) The program may not terminate.

(D)。**程序可终止**，ranking function $21 \times X + 2 \times (111 - N)$。

1-3

参考解答

Question

Let $[X = 0]$ while b do c end $[X = 1]$ be a Hoare triple. Which of the following is *true*?

(A) If c is $X := 1$, then the Hoare triple is valid for some b.

(B) If b is true, then the Hoare triple is valid for some c.

(C) If b is $X \neq 1$, then the Hoare triple is valid no matter what c is.

(D) The Hoare triple is always invalid no matter what b and c are.

(A)。其中 b 取 $X = 0$。

Question

Recall that two IMP programs (with havoc) c_1 and c_2 are *behaviorally equivalent*, if for every states σ and σ', their big-step operational semantic evaluation relations satisfy $\langle \sigma, c_1 \rangle \Downarrow \sigma' \iff \langle \sigma, c_2 \rangle \Downarrow \sigma'$. In which of the following are c_1 and c_2 behaviorally equivalent?

(A) $c_1 : X := Y; Y := X \qquad c_2 : Y := X; X := Y$

(B) $c_1 : \text{skip} \qquad c_2 : \text{if } X > 10 \text{ then } X := 0 \text{ else skip fi}$

(C) $c_1 : \text{havoc } X; X := 10 \qquad c_2 : X := 10$

(D) $c_1 : \text{havoc } X; \text{havoc } Y \qquad c_2 : \text{havoc } Y$

(C)。直观：$X = 10$ 覆盖了 havoc X 对 X 的随机赋值操作。

1-5
参考解答

Question

Let F be a CNF with four variables x_1, x_2, x_3, x_4. We apply the DPLL algorithm (without backjump) on F and the following operations are done: decide x_1, propagate x_2, propagate x_3. Which of the following operations will be possibly done in the next step?

(A) Decide $\neg x_3$.

(B) Backtrack and decide $\neg x_1$.

(C) Backtrack and decide $\neg x_2$.

(D) Backtrack and decide $\neg x_3$.

(B)。在没有 backjump 时，回溯操作将回到最近一次 decide（而非 propagate）。

Contents

2-1

Question

We consider ...

> If an assertion statement fails, it causes the program to go into an *error state* and exit (or abort).
>
> If an assumption statement fails, the program fails to evaluate at all. In other words, the program gets *stuck* and has no final state.

... "$\langle \sigma, c \rangle \Downarrow r$", where the evaluation *result*

$$r ::= \mathrm{norm}(\sigma) \mid \mathrm{err}$$

can state two possible cases: $\mathrm{norm}(\sigma)$ for normally execution with ending state σ, or err for reaching the error state ...
Give the evaluation rules for assumption and assertion statements.

2-1
参考解答

$$(\text{AssertionTrue})\,\frac{\mathcal{B}[\![b]\!]_\sigma = \top}{\langle \sigma, \textsf{assert}\ b\rangle \Downarrow \textsf{norm}(\sigma)} \qquad (\text{AssertionFalse})\,\frac{\mathcal{B}[\![b]\!]_\sigma = \bot}{\langle \sigma, \textsf{assert}\ b\rangle \Downarrow \textsf{err}}$$

$$(\text{AssumptionTrue})\,\frac{\mathcal{B}[\![b]\!]_\sigma = \top}{\langle \sigma, \textsf{assume}\ b\rangle \Downarrow \textsf{norm}(\sigma)}$$

2-1
错误解答

$$(\text{AssumeFalse})\ \frac{\mathcal{B}[\![b]\!]_\sigma = \bot}{\langle \sigma, \textsf{assume } b \rangle \Downarrow \textsf{err}} \qquad (\text{AssumptionFalse})\ \frac{\mathcal{B}[\![b]\!]_\sigma = \bot}{\langle \sigma, \textsf{assume } b \rangle \Downarrow \bot}$$

$$(\text{Assumption})\ \frac{\mathcal{B}[\![b \to b']\!]_\sigma = \top}{\langle \sigma, \textsf{assume } b \rangle \Downarrow \textsf{norm}(\sigma) \quad \mathcal{B}[\![b']\!]_\sigma = \top}$$

注意题干：If an assumption statement fails, the program fails to evaluate at all. In other words, the program gets *stuck* and has no final state.

$$(\text{AssumptionTrue})\,\frac{\mathcal{B}[\![b]\!]_\sigma = \top}{\langle \sigma, \textsf{assume}\ b\rangle \Downarrow \sigma}$$

注意符号的规范性！

2-1

不规范解答 2

$$(\text{assumeErr})\ \frac{\mathcal{B}[\![b]\!]_\sigma = \perp}{\langle \sigma, \textsf{assert}\ b \rangle \Downarrow}$$

Stuck 的规则无需写出！

2-2
错误解答 1

Question

… We redefine Hoare triples "$\{P\}\ c\ \{Q\}$" to mean that, whenever c is started in a state satisfying P, and terminates with result r, then $r = \text{norm}(\sigma)$ (and hence $r \neq \text{err}$) where the state σ satisfies Q.
Design Hoare rules for assumption and assertion statements.

$$(\text{Assert})\,\frac{}{\{P\}\ \textsf{assert}\ \mathrm{b}\ \{P\}} \qquad (\text{AssertFalse})\,\frac{P \not\Rightarrow b}{\{P\}\ \textsf{assert}\ b\ \{\bot\}}$$

反例： assert false 的终止状态为 err。

错误解答 2

$$\text{Assert}\ \frac{\{P \wedge b\}\textsf{assert}\ b; C\{Q\}}{\{P\}C\{Q\}}$$

$$\text{AssertErr}\ \frac{\{P \wedge \neg b\}\textsf{assert}\ b; C\{Q\}}{\textsf{err}}$$

$$\text{Assume}\ \frac{\{P\}\textsf{assume}\ b; C\{Q\}}{\{P\}C\{Q\}}$$

两处错误：(1) 前提和结论写反；(2) 默认 assertion/assumption 语句后面还有语句。

2-2
弱化版本 1

$$\text{Assume}\ \frac{P \ \wedge \ b}{\{P\}\ \textsf{assume}\ b\ \{P \ \wedge \ b\}} \qquad (\text{Assume})\ \frac{P \wedge b \Rightarrow \top}{\{P\}\textsf{assume}\ b\{P \wedge b\}}$$

忽略了 $(P \wedge b)$ 不是永真式的情况。

弱化版本 2

$$(\text{Assume})\frac{P \Rightarrow b}{\{P\}\textsf{assume } b\{P\}}$$

$$(\text{Assert})\frac{P \Rightarrow b}{\{P\}\textsf{assert } b\{P\}}$$

$$\{P \wedge b\} \textsf{ assume } b \{P \wedge b\}$$

$$\{\neg b\} \textsf{ assume } b \{false\}$$

$$(\text{AssertTrue})\frac{P \Rightarrow b}{\{P\} \textsf{ assert } b \{P\}}$$

$$(\text{AssertFalse})\frac{P \Rightarrow \neg b}{\{P\} \textsf{ assume } b \{\top\}}$$

$$(\text{AssumeTrue})\frac{P \Rightarrow b}{\{P\} \textsf{ assume } b \{P\}}$$

$$(\text{AssumeFalse})\frac{P \Rightarrow \neg b}{\{P\} \textsf{ assume } b \{\bot\}}$$

忽略了 $\neg(P \Rightarrow b) \wedge \neg(P \Rightarrow \neg b)$ 的情况，例如

$$\{X > 0\} \textsf{ assume } Y > 0 \{X > 0 \wedge Y > 0\}$$

2-2

参考解答

$$(\text{Assertion})\ \frac{}{\{b \wedge P\}\ \textsf{assert}\ b\ \{P\}}$$

$$(\text{Assumption})\ \frac{}{\{b \rightarrow P\}\ \textsf{assume}\ b\ \{P\}}$$

2-2
不规范解答

$$\text{(Assumption)}\ \frac{}{\{b \mapsto P\}\textsf{assume}\ b\{P\}}$$

$$\text{(assertErr)}\ \frac{P \nRightarrow b}{\{P\}\textsf{assert}\ b\{\mathsf{err}\}}$$

$$\text{(assumeErr)}\ \frac{P \nRightarrow b}{\{P\}\textsf{assume}\ b\{\}}$$

Solution $\{\mathcal{B}[\![b]\!]_P = \textsf{true}\}\textsf{assert}\ b\{P\}$

$\{\mathcal{B}[\![b]\!]_P = \textsf{true}\}\textsf{assume}\ b\{P\}$ ■

注意符号的规范性!

2-3
参考解答

Question

Compute wlp($X := X + 1$; assume $X > 0$; $Y := Y + X, X + Y + Y \geq 3$).

$$
\begin{aligned}
&\text{wlp}(X := X + 1; \text{assume } X > 0; Y := Y + X, X + Y + Y \geq 3) \\
=&\text{wlp}(X := X + 1; \text{assume } X > 0, \text{wlp}(Y := Y + X, X + Y + Y \geq 3)) \\
=&\text{wlp}(X := X + 1; \text{assume } X > 0, X + Y + X + Y + X \geq 3) \\
=&\text{wlp}(X := X + 1, \text{wlp}(\text{assume } X > 0, X + Y + X + Y + X \geq 3)) \\
=&\text{wlp}(X := X + 1, X > 0 \to X + Y + X + Y + X \geq 3) \\
=&X + 1 > 0 \to X + 1 + Y + X + 1 + Y + X + 1 \geq 3 \\
=&X + 1 > 0 \to 3X + 2Y + 3 \geq 3 \\
=&X + 1 \leq 0 \vee 3X + 2Y \geq 0
\end{aligned}
$$

2-3

错误解答 1

$$
\begin{aligned}
&\mathsf{wlp}(X := X+1; \textsf{assume}\ X > 0; Y := Y + X, X + Y + Y \geq 3) \\
=&\mathsf{wlp}(X := X+1; \textsf{assume}\ X > 0, 2 \times Y + 3 \times X \geq 3) \\
=&\mathsf{wlp}(X := X+1, X > 0 \wedge 2 \times Y + 3 \times X \geq 3) \\
=&X + 1 > 0 \wedge 2 \times Y + 3 \times (X+1) \geq 3 \\
=&X + 1 > 0 \wedge 2 \times Y + 3 \times X \geq 0
\end{aligned}
$$

规则： $\mathsf{wlp}(\textsf{assume}\ b, Q) = b \to Q$， **而不是** $\mathsf{wlp}(\textsf{assume}\ b, Q) = b \wedge Q$!

2-3
错误解答 2

$$= X + 1 > 1 \rightarrow X + 1 + Y + X + 1 + Y + X + 1 \geq 3$$

$$= \top$$

?

画蛇添足，最后一步的等号不成立！

2-3

不规范解答

$$\mathsf{wlp}(X := X + 1; \textsf{assume } X > 0; Y := Y + X, X + Y + Y \geq 3)$$
$$\mathsf{wlp}(X := X + 1; \textsf{assume } X > 0, X + Y + X + Y + X \geq 3)$$
$$\mathsf{wlp}(X := X + 1, (X > 0) \rightarrow 3X + 2Y \geq 3)$$
$$X + 1 > 0 \rightarrow 3(X + 1) + 2Y \geq 3$$
$$X + 1 > 0 \rightarrow 3X + 2Y \geq 0$$

计算过程用等号连接起来!

Contents

后半期主要内容

- 程序语义（直观 + 形式系统）
  - Big-step operational semantics
  - Small-step operational semantics (stuck, normal form, deterministic, etc.)
- 程序验证
  - 正确性刻画： Hoare triples
  - 证明方法
    - 按 Hoare triples 有效性的定义（需要直接用到程序语义）
    - 按 Hoare rules 进行演绎推理，过程用 decorated program 表示
    - 计算 wlp，看给定的前置条件是否蕴含该最弱前置条件
  - 难点： 循环
    - 寻找合适的循环不变式来证明程序的正确性
    - 寻找合适的秩函数来证明程序的终止性

理论概念

partial correctness v.s. total correctness

valid v.s. invalid (Hoare triples)

soundness v.s. completeness (of a proof system)

weak v.s. strong

assumption v.s. assertion

复习（预习）建议

以概念辨析和算法（或形式系统）原理的理解为主
优先复习课程中反复提到的概念和方法
优先复习书面作业中涉及的概念、方法、题型