ECNU

SEI

程序验证方法

研究生课程

Chapter 3 (3.3) while Programs

朱惠彪

华东师范大学 软件工程学院

ECNU
SEI

3.3 Verification

- **Correctness formulas**
  - **{p} S {q}**
  - **S is a while program and p and q are assertions.**
- **Partial correctness (部分正确性)**
  - **If every terminating computation of S that starts in a state satisfying p terminates in a state satisfying q.**

对程序**S**的任何一个终止计算，若程序**S**在开始时满足**p**，那么**S**在终止时满足**q**。

- **Total correctness (完全正确性)**
  - **If every computation of S that starts in a state satisfying p terminates and its final state satisfies q.**

对程序**S**的在开始时满足**p**的任何一个计算，**S**能成功终止且终止时满足**q**。

ECNU

SEI

Review

Definition 3.2. We now define two input/output semantics for **while** programs. Each of them associates with a program S and a proper state $\sigma \in \Sigma$ a set of output states.

(i) The *partial correctness semantics* is a mapping

$$\mathcal{M}[\![S]\!] : \Sigma \rightarrow \mathcal{P}(\Sigma)$$

with

$$\mathcal{M}[\![S]\!](\sigma) = \{\tau \mid < S, \sigma > \ \rightarrow^* \ < E, \tau >\}.$$

(ii) The *total correctness semantics* is a mapping

$$\mathcal{M}_{tot}[\![S]\!] : \Sigma \rightarrow \mathcal{P}(\Sigma \cup \{\perp\})$$

with

$$\mathcal{M}_{tot}[\![S]\!](\sigma) = \mathcal{M}[\![S]\!](\sigma) \cup \{\perp \mid S \text{ can diverge from } \sigma\}.$$

ECNU

SEI

Definition 3.3

(i) We say that the correctness formula $\{p\}\ S\ \{q\}$ is true in the sense of *partial correctness*, and write $\models \{p\}\ S\ \{q\}$, if

$$\mathcal{M}[\![S]\!]([\![p]\!]) \subseteq [\![q]\!].$$

(ii) We say that the correctness formula $\{p\}\ S\ \{q\}$ is true in the sense of *total correctness*, and write $\models_{tot} \{p\}\ S\ \{q\}$, if

$$\mathcal{M}_{tot}[\![S]\!]([\![p]\!]) \subseteq [\![q]\!].$$

□

Note 1: $\perp \notin$ [[q]] (page 64)

Note 2:

(1) Correctness formula {p} S {q} is true in the sense of partial correctness if every terminating computation of S that starts in a state satisfying p terminates in a state satisfying q.

(2) {p} S {q} is true in the sense of total correctness if every computation of S that starts in a state satisfying p terminates and its final state satisfies q.

(3)Thus in the case of partial correctness, diverging computations of S are not taken into account.

ECNU SEI

Example 3.2.

- $S \equiv a[0] := 1; a[1] := 0;$ **while $a[x] \neq 0$ do $x := x + 1$ od**
- **Correctness formulas**
  - 1、$\{x = 0\}\ S\ \{a[0] = 1 \land a[1] = 0\}$
  - 2、$\{x = 0\}\ S\ \{x = 1 \land a[x] = 0\}$
  - 3、$\{x = 2\}\ S\ \{true\}$
  - 4、$\{x = 2 \land \forall i \geq 2 : a[i] = 1\}\ S\ \{false\}$
- **Total correctness**
  - 1、2
- **Partial correctness**
  - 1、2、3、4

Let τ be a state in which x is 2 and for i = 2, 3, . . ., a[i] is 1 (how about 0?). Consider S starting in τ

ECNU

SEI

Partial Correctness

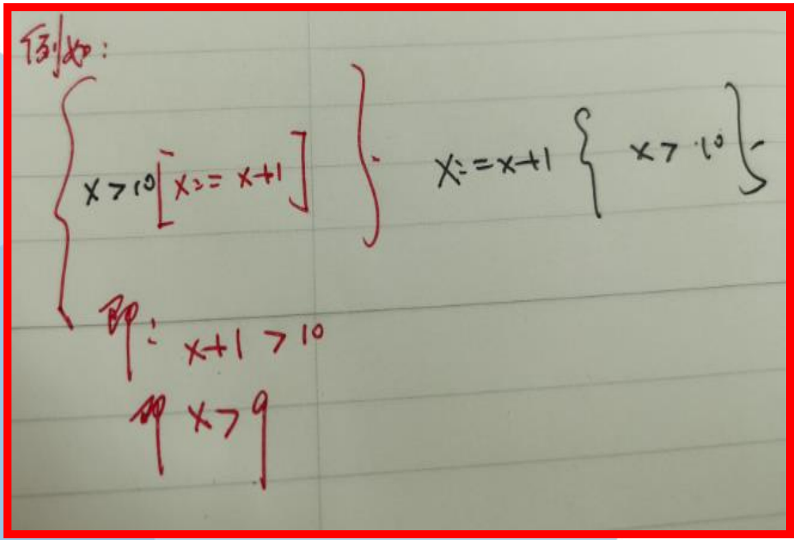
- **Axiom 1: Skip**

$$\{p\}\ skip\ \{p\}$$

- **Axiom 2: Assignment**

$$\{p[u := t]\}\ u := t\ \{p\}$$

- **Axiom 3: Composition**



$$\frac{\{p\}\ S_1\ \{r\}, \{r\}\ S_2\ \{q\}}{\{p\}\ S_1;\ S_2\ \{q\}}$$

ECNU

SEI

Partial Correctness

- **Rule 4: Conditional**

$$\frac{\{p \wedge B\}\ S_1\ \{q\}, \{p \wedge \neg B\}\ S_2\ \{q\}}{\{p\}\ \textbf{if}\ B\ \textbf{then}\ S_1\ \textbf{else}\ S_2\ \textbf{fi}\ \{q\}}$$

- **Rule 5: Loop**

P----循环不变式 (Loop Invariant）

$$\frac{\{p \wedge B\}\ S\ \{p\}}{\{p\}\ \textbf{while}\ B\ \textbf{do}\ S\ \textbf{od}\ \{p \wedge \neg B\}}$$

- **Rule 6: Consequence**

$$\frac{p \rightarrow p_1, \{p_1\}\ S\ \{q_1\}, q_1 \rightarrow q}{\{p\}\ S\ \{q\}}$$

Partial Correctness

- **PROOF SYSTEM PW :**

  This system consists of the group of axioms and rules 1-6.

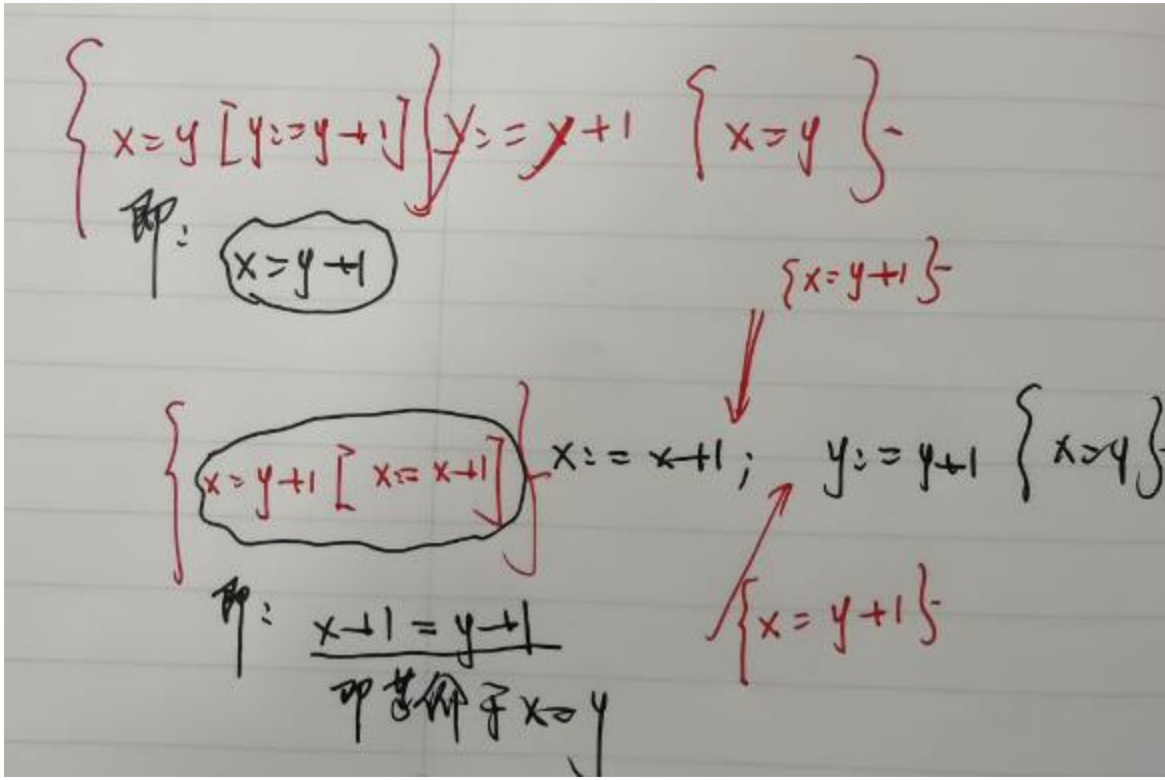
ECNU SEI

Example 3.3.(i) (page 66, 67)

- **Consider the program:**
  - S ≡ x := x + 1; y := y + 1
- **Prove in the system PW the correctness formula:**
  - {x = y} S {x = y}

$$\{p[u := t]\}\ u := t\ \{p\}$$

$$\frac{\{p\}\ S_1\ \{r\},\{r\}\ S_2\ \{q\}}{\{p\}\ S_1;\ S_2\ \{q\}}$$



ECNU

SEI

Example 3.3.(i): Proof

The program:
$S \equiv x := x + 1; y := y + 1$

The correctness formula:
$\{x = y\}\ S\ \{x = y\}$

AXIOM 2: ASSIGNMENT

$\{p[u := t]\}\ u := t\ \{p\}$

- $y := y + 1$
  - Apply Axiom 2: Assignment and backward substitution:
    - $\{x = y\ [y := y + 1]\}\ y := y + 1\ \{x = y\}$
    - $\{x = y + 1\}\ y := y + 1\ \{x = y\}$
- $x := x + 1$
  - Apply Axiom 2: Assignment and backward substitution:
    - $\{x = y + 1\ [x := x + 1]\}\ x := x + 1\ \{x = y + 1\}$
    - $\{x + 1 = y + 1\}\ x := x + 1\ \{x = y + 1\}$
- S
  - Apply Rule 3: Composition
    - $\{x + 1 = y + 1\}\ x := x + 1; y := y + 1\ \{x = y\}$
  - Appley Rule 6: Consequence
    - $x = y \rightarrow x + 1 = y + 1$

ECNU

SEI

Example 3.4

Assume：x=22， y=5

rem	quo
22	0
17	1
12	2
7	3
2	4

For each time,
Please investigate:
quo.y+rem=x (=22)

- **Consider the following program DIV for computing the quotient and remainder of two natural numbers x and y:**
  - **DIV ≡ quo := 0; rem := x; S0**
  - **S0 ≡ while rem ≥ y do rem := rem − y; quo := quo + 1 od**

In the system PW, we wish to prove:

$$\{x \geq 0 \wedge y \geq 0\}\ \mathrm{DIV}\ \{quo \cdot y + rem = x \wedge 0 \leq rem < y\}$$

Example 3.4: Proof (1/5)

ECNU

SEI

- **Loop invariant of S0:**
  - $p \equiv quo \cdot y + rem = x \land rem \geq 0$

- **Prove the following three facts:**

(1) $\{x \geq 0 \land y \geq 0\}$ quo := 0; rem := x $\{p\}$

(2) $\{p \land rem \geq y\}$ rem := rem − y; quo := quo + 1 $\{p\}$

(3) $p \land \neg(rem \geq y) \rightarrow quo \cdot y + rem = x \land 0 \leq rem < y$ (Clear)

Assume: x=22, y=5

rem	quo
22	0
17	1
12	2
7	3
2	4

For each time, Please investigate: quo.y+rem=x (=22)

$$\frac{\{p \wedge B\}\ S\ \{p\}}{\{p\}\ \text{while } B \text{ do } S \text{ od } \{p \wedge \neg B\}}$$

The program:

DIV ≡ quo := 0; rem := x; S0

S0 ≡ while rem ≥ y do rem := rem − y ; quo := quo + 1 od

The correctness formula:

$\{x \geq 0 \land y \geq 0\}$ DIV $\{quo \cdot y + rem = x \land 0 \leq rem < y\}$

ECNU

SEI

Example 3.4: Proof (2/5)

$\{x \geq 0 \wedge y \geq 0\}$
quo := 0; rem := x
$\{p\}$

$p \equiv quo \cdot y + rem = x \wedge rem \geq 0$

- **rem := x**
  - **Apply Axiom 2: Assignment**
    - $\{p\ [rem := x]\}$ rem := x $\{p\}$
    - $\{quo \cdot y + x = x \wedge x \geq 0\}$ rem := x $\{p\}$
- **quo := 0**
  - **Apply Axiom 2: Assignment**
    - $\{quo \cdot y + x = x \wedge x \geq 0\ [quo := 0]\}$ quo := 0 $\{quo \cdot y + x = x \wedge x \geq 0\}$
    - $\{0 \cdot y + x = x \wedge x \geq 0\}$ quo := 0 $\{quo \cdot y + x = x \wedge x \geq 0\}$
- **quo := 0; rem := x**
  - **Apply Rule 3: Composition**
    - $\{0 \cdot y + x = x \wedge x \geq 0\}$ quo := 0; rem := x $\{p\}$
  - **Apply Rule 6: Consequence**
    - $x \geq 0 \wedge y \geq 0 \rightarrow 0 \cdot y + x = x \wedge x \geq 0$

ECNU

SEI

Example 3.4: Proof (3/5)

$\{p \wedge rem \geq y\}$ $rem := rem - y$; $quo := quo + 1$ $\{p\}$

$p \equiv quo \cdot y + rem = x \wedge rem \geq 0$

- $quo := quo + 1$
  - Apply Axiom 2: Assignment
    - $\{p\,[quo := quo + 1]\}$ $quo := quo + 1$ $\{p\}$
    - $\{(quo + 1) \cdot y + rem = x \wedge rem \geq 0\}$ $quo := quo + 1$ $\{p\}$
- $rem := rem - y$
  - Apply Axiom 2: Assignment
    - $\{\{(quo + 1) \cdot y + rem = x \wedge rem \geq 0\ [rem := rem - y]\}$ $rem := rem - y$ $\{(quo + 1) \cdot y + rem = x \wedge rem \geq 0\}$
    - $\{(quo + 1) \cdot y + (rem - y) = x \wedge rem - y \geq 0\}$ $rem := rem - y$ $\{(quo + 1) \cdot y + rem = x \wedge rem \geq 0\}$

ECNU

SEI

Example 3.4: Proof (4/5)

$\{p \wedge rem \geq y\}$ $rem := rem - y; quo := quo + 1$ $\{p\}$

$p \equiv quo \cdot y + rem = x \wedge rem \geq 0$

- $rem := rem - y; quo := quo + 1$
  - Apply Rule 3: Composition
    - $\{(quo + 1) \cdot y + (rem - y) = x \wedge rem - y \geq 0\} rem := rem - y; quo := quo + 1 \{p\}$
  - Apply Rule 6: Consequence
    - $p \wedge rem \geq y \rightarrow (quo + 1) \cdot y + (rem - y) = x \wedge rem - y \geq 0$

Example 3.4: Proof (5/5)

ECNU SEI

The program:
DIV ≡ quo := 0; rem := x; S0
S0 ≡ while rem ≥ y do rem := rem − y; quo := quo + 1 od

The correctness formula:
$\{x \geq 0 \wedge y \geq 0\}$ DIV $\{quo \cdot y + rem = x \wedge 0 \leq rem < y\}$

p ≡ quo · y + rem = x ∧ rem ≥ 0

- $\{p \wedge rem \geq y\}$ rem := rem − y; quo := quo + 1 $\{p\}$
  - Apply Rule 5: Loop
    - $\{p\}$ S0 $\{p \wedge \neg(rem \geq y)\}$
- $\{x \geq 0 \wedge y \geq 0\}$ quo := 0; rem := x $\{p\}$
  - Apply Rule 3: Composition
    - $\{x \geq 0 \wedge y \geq 0\}$ DIV $\{p \wedge \neg(rem \geq y)\}$
- $p \wedge \neg(rem \geq y) \rightarrow quo \cdot y + rem = x \wedge 0 \leq rem < y$
  - Apply Rule 6: Cosequence
    - $\{x \geq 0 \wedge y \geq 0\}$ DIV $\{quo \cdot y + rem = x \wedge 0 \leq rem < y\}$

$$\frac{\{p \wedge B\}\ S\ \{p\}}{\{p\}\ \textbf{while}\ B\ \textbf{do}\ S\ \textbf{od}\ \{p \wedge \neg B\}}$$

ECNU

SEI

Total Correctness

- **Rule 7: Loop II**

$$\frac{\begin{array}{l}\{p \wedge B\}\ S\ \{p\},\\ \{p \wedge B \wedge t = z\}\ S\ \{t < z\},\\ p\ \rightarrow t \geq 0\end{array}}{\{p\}\ \textbf{while}\ B\ \textbf{do}\ S\ \textbf{od}\ \{p \wedge \neg\ B\}}$$

  - **t is an integer expression and z is an integer variable that does not appear in p, B, t or S.**

- **The second premise: z holds the initial value of t and t is decreased with each iteration.**
- **The third premise: t is nonnegative if another iteration can be performed.**
- **Thus no infinite computation is possible.**
- **Expression t is called a bound function of the loop while B do S od.**

ECNU

SEI

Total Correctness

- **PROOF SYSTEM TW :**

  This system consists of the group of axioms and rules 1-4, 6, 7.

ECNU

SEI

Example 3.5

Assume: x=22, y=5

rem	quo
22	0
17	1
12	2
7	3
2	4

For each time,
Please investigate:
quo.y+rem=x (=22)

- **Consider the following program DIV for computing the quotient and remainder of two natural numbers x and y:**
  - **DIV ≡ quo := 0; rem := x; S0**
  - **S0 ≡ while rem ≥ y do rem := rem − y; quo := quo + 1 od**
- Prove in the system TW the correctness formula

$$\{x \geq 0 \wedge y > 0\}\ \mathrm{DIV}\ \{quo \cdot y + rem = x \wedge 0 \leq rem < y\}$$

ECNU

SEI

Example 3.5： Proof (1/3)

The three facts in example 3.4:

$\{x \geq 0 \wedge y \geq 0\}$ quo := 0; rem := x $\{p\}$

$\{p \wedge rem \geq y\}$ rem := rem − y; quo := quo + 1 $\{p\}$

$p \wedge \neg(rem \geq y) \rightarrow quo \cdot y + rem = x \wedge 0 \leq rem < y$

DIV ≡ quo := 0; rem := x; S0

```
  S0
≡ while rem ≥ y do
    rem := rem − y;
    quo := quo + 1
  od
```

$$\frac{\{p \wedge B\}\ S\ \{p\},\quad \{p \wedge B \wedge t = z\}\ S\ \{t < z\},\quad p \rightarrow t \geq 0}{\{p\}\ \textbf{while}\ B\ \textbf{do}\ S\ \textbf{od}\ \{p \wedge \neg B\}}$$

- The assertion p is the loop invariant of S0 in example 3.4:
  - $p \equiv quo \cdot y + rem = x \wedge rem \geq 0$
- Let **p′ be the loop invariant** and let t be the bound function.
  - $p' \equiv p \wedge y > 0$
  - $t \equiv rem$
- Prove the following five facts:

(1) $\{x \geq 0 \wedge y > 0\}$ quo := 0; rem := x $\{p'\}$

(2) $\{p' \wedge rem \geq y\}$rem := rem − y;quo := quo + 1 $\{p'\}$

(3) $\{p' \wedge rem \geq y \wedge \mathbf{rem = z}\}$ rem := rem − y; quo := quo + 1$\{\mathbf{rem < z}\}$

(4) $p' \rightarrow \mathbf{rem \geq 0}$ (Clear)

(5) $p' \wedge \neg(rem \geq y) \rightarrow quo \cdot y + rem = x \wedge 0 \leq rem < y$

ECNU

SEI

Example 3.5: Proof (2/3)

{p′ ∧ rem ≥ y ∧ rem = z} rem := rem − y; quo := quo + 1 {rem < z}

p′ ≡ p ∧ y > 0

p ≡ quo · y + rem = x ∧ rem ≥ 0

$$\frac{\{p \wedge B\}\ S\ \{p\},\quad \{p \wedge B \wedge t = z\}\ S\ \{t < z\},\quad p \rightarrow t \geq 0}{\{p\}\ \text{while}\ B\ \text{do}\ S\ \text{od}\ \{p \wedge \neg B\}}$$

- **quo := quo + 1**
  - Apply Axiom 2: Assignment
    - {rem < z [quo := quo + 1]} quo := quo + 1{rem < z}
    - {rem < z} quo := quo + 1{rem < z}
- **rem := rem − y**
  - Apply Axiom 2: Assignment
    - {rem < z [rem := rem - y]}rem := rem − y{rem < z}
    - {(rem – y) < z}rem := rem − y{rem < z}
- **rem := rem − y; quo := quo + 1**
  - Apply Rule 3: Composition
    - {**(rem – y)** < z}**rem := rem − y; quo := quo + 1**{rem < z}
  - Apply Rule 6: Consequence
    - **p ∧ y > 0 ∧ rem ≥ y ∧ rem = z → (rem − y) < z**

ECNU

SEI

Example 3.5: Proof (3/3)

$$\frac{\{p \wedge B\}\ S\ \{p\},\quad \{p \wedge B \wedge t = z\}\ S\ \{t < z\},\quad p \rightarrow t \geq 0}{\{p\}\ \text{while}\ B\ \text{do}\ S\ \text{od}\ \{p \wedge \neg B\}}$$

DIV ≡ quo := 0; rem := x; S0

S0

≡ while rem ≥ y do

rem := rem − y;

quo := quo + 1

od

p′ ≡ p ∧ y > 0

p ≡ quo · y + rem = x ∧ rem ≥ 0

- $\{p' \wedge \mathbf{rem} \geq \mathbf{y}\}$ rem := rem − y; quo := quo + 1 $\{p'\}$
- $\{p' \wedge \mathbf{rem} \geq \mathbf{y} \wedge rem = z\}$ rem := rem − y; quo := quo + 1 $\{rem < z\}$
- $\mathbf{p' \rightarrow rem \geq 0}$
  - Apply Rule 7: Loop II
    - $\{p'\}$ **S0** $\{p' \wedge \neg(\mathbf{rem} \geq \mathbf{y})\}$
- $\{\mathbf{x \geq 0 \wedge y > 0}\}$ **quo := 0; rem := x** $\{p'\}$
  - Apply Rule 3: Composition
    - $\{\mathbf{x \geq 0 \wedge y > 0}\}$ **DIV** $\{p' \wedge \neg(\mathbf{rem} \geq \mathbf{y})\}$
- $p' \wedge \neg(rem \geq y) \rightarrow \mathbf{quo \cdot y + rem = x \wedge 0 \leq rem < y}$
  - Apply Rule 6: Cosequence
    - $\{\mathbf{x \geq 0 \wedge y > 0}\}$ **DIV** $\{\mathbf{quo \cdot y + rem = x \wedge 0 \leq rem < y}\}$

ECNU

SEI

Decomposition

RULE A1: DECOMPOSITION

$$\frac{\vdash_p \{p\}\ S\ \{q\},\quad \vdash_t \{p\}\ S\ \{\mathbf{true}\}}{\{p\}\ S\ \{q\}}$$

ECNU

SEI

Soundness

The program:

DIV

≡ quo := 0; rem := x;

S0

S0

≡ while rem ≥ y do

rem := rem − y;

quo := quo + 1

od

We have just established

- $\vdash_{PW} \{x \geq 0 \wedge y \geq 0\}$ DIV $\{quo \cdot y + rem = x \wedge 0 \leq rem < y\}$
- and
- $\vdash_{TW} \{x \geq 0 \wedge y > 0\}$ DIV $\{quo \cdot y + rem = x \wedge 0 \leq rem < y\}$

However, our goal was to show

- $\models \{x \geq 0 \wedge y \geq 0\}$ DIV $\{quo \cdot y + rem = x \wedge 0 \leq rem < y\}$
- and
- $\models_{tot} \{x \geq 0 \wedge y > 0\}$ DIV $\{quo \cdot y + rem = x \wedge 0 \leq rem < y\}$

ECNU

SEI

Definition 3.4

Definition 3.3 (P64)

$\models \{p\}\ S\ \{q\}$ if $\mathcal{M}[[S]]([[p]]) \subseteq [[q]]$.

$\models_{tot} \{p\}\ S\ \{q\}$ if $\mathcal{M}_{tot}[[S]]([[p]]) \subseteq [[q]]$.

Let G be a proof system allowing us to prove correctness formulas about programs in a certain class C. **We say that G is *sound for partial correctness*** *of programs in C* if for all correctness formulas $\{p\}\ S\ \{q\}$ about programs in C

$$\vdash_G \{p\}\ S\ \{q\} \text{ implies } \underline{\models \{p\}\ S\ \{q\}},$$

and **G is *sound for total correctness*** *of programs in C* if for all correctness formulas $\{p\}\ S\ \{q\}$ about programs in C

$$\vdash_G \{p\}\ S\ \{q\} \text{ implies } \underline{\models_{tot} \{p\}\ S\ \{q\}}.$$

When the class of programs C is clear from the context, we omit the reference to it.

Theorem 3.1. (Soundness of PW and TW)

(i) ***The proof system PW is sound for partial correctness*** *of* **while** *programs.*

(ii) ***The proof system TW is sound for total correctness*** *of* **while** *programs.*

Due to the form of the proof systems *PW* and *TW*, ***it is sufficient to prove that all axioms of PW (TW) are true in the sense of partial (total) correctness and that all proof rules of PW (TW) are sound for partial (total) correctness.*** Then the result follows by the induction on the length of proofs.

We consider all axioms and proof rules in turn.

ECNU

SEI

SKIP

AXIOM 1: SKIP

{p} skip {p}

- Clearly
- $N[[skip]]([[p]]) = [[p]]$
- for any assertion *p*, so the skip axiom is true in the sense of partial (total) correctness.

ECNU
SEI

ASSIGNMENT

AXIOM 2: ASSIGNMENT
{p[u := t]} u := t {p}

- Let p be an assertion. By the Substitution Lemma 2.4 and transition axiom (ii), whenever $\mathbf{N[[u := t]](\sigma) = \{\tau\}}$, then

$$\sigma \models p[u := t] \text{ iff } \tau \models p.$$

- This implies $\mathbf{N[[u := t]]([[p[u := t]]]) \subseteq [[p]]}$, so the assignment axiom is true in the sense of partial (total) correctness.

Lemma 2.4. (Substitution)
(i) $\sigma(s[u := t]) = \sigma[u := \sigma(t)](s)$,
(ii) $\sigma \models p[u := t]$ *iff* $\sigma[u := \sigma(t)] \models p$.

transition axioms and rules
(ii) $< u := t, \sigma > \; \rightarrow \; < E, \sigma[u := \sigma(t)] >$

$$\underbrace{x \geq \boxed{9}}_{p[x := x+1]}$$

$$\{p[x := x+1]\}\ x := x+1\ \{\underbrace{x \geq \boxed{10}}_{p}\}$$

数据状态：$\sigma: x \rightarrow 13$　　数据状态：$\tau: x \rightarrow 14$

有 $\sigma \models p[x := x+1]$ iff $\tau \models p$

ECNU

SEI

COMPOSITION

RULE 3: COMPOSITION

$$\frac{\{p\}\ S_1\ \{r\},\ \{r\}\ S_2\ \{q\}}{\{p\}\ S_1;\ S_2\ \{q\}}$$

- Suppose that

$$N[[S_1]]([[p]]) \subseteq [[r]]$$

and

$$N[[S_2]]([[r]]) \subseteq [[q]].$$

- Then by the monotonicity of $N[[S_2]]$ (the Input/Output Lemma 3.3(i))

$$N[[S_2]](N[[S_1]]([[p]])) \subseteq N[[S_2]]([[r]]) \subseteq [[q]].$$

- But by the Input/Output Lemma 3.3(ii)

$$N[[S_1;\ S_2]]([[p]]) = N[[S_2]](N[[S_1]]([[p]]));$$

so

$$N[[S_1;\ S_2]]([[p]]) \subseteq [[q]].$$

- Thus the composition rule is sound for partial (total) correctness.

Lemma 3.3. (Input/Output)

(i) $\mathcal{N}[\![S]\!]$ *is monotonic; that is,* $X \subseteq Y \subseteq \Sigma \cup \{\bot\}$ *implies* $\mathcal{N}[\![S]\!](X) \subseteq \mathcal{N}[\![S]\!](Y)$.

(ii) $\mathcal{N}[\![S_1;\ S_2]\!](X) = \mathcal{N}[\![S_2]\!](\mathcal{N}[\![S_1]\!](X))$.

ECNU

SEI

CONDITIONAL

RULE 4: CONDITIONAL

$$\frac{\{p \wedge B\}\ S_1\ \{q\},\ \{p \wedge \neg B\}\ S_2\ \{q\}}{\{p\}\ \text{if } B \text{ then } S_1 \text{ else } S_2 \text{ fi}\ \{q\}}$$

- Suppose that

$$N[[S_1]]([[p \wedge B]]) \subseteq [[q]]$$

and

$$N[[S_2]]([[p \wedge \neg B]]) \subseteq [[q]].$$

- By the Input/Output Lemma 3.3(iv)

$$N[[\text{if } B \text{ then } S_1 \text{ else } S_2 \text{ fi}]]([[p]])$$

$$= N[[S_1]]([[p \wedge B]]) \cup N[[S_2]]([[p \wedge \neg B]]);$$

so

$$N[[\text{if } B \text{ then } S_1 \text{ else } S_2 \text{ fi}]]([[p]]) \subseteq [[q]].$$

- Thus the conditional rule is sound for partial (total) correctness.

Lemma 3.3. (Input/Output)

(iv) $\mathcal{N}[\![\text{if } B \text{ then } S_1 \text{ else } S_2 \text{ fi}]\!](X) =$
$\mathcal{N}[\![S_1]\!](X \cap [\![B]\!]) \cup \mathcal{N}[\![S_2]\!](X \cap [\![\neg B]\!]) \cup \{\bot \mid \bot \in X \textit{ and } \mathcal{N} = \mathcal{M}_{tot}\}$

ECNU

SEI

LOOP

RULE 5: LOOP

$$\frac{\{p \wedge B\}\ S\ \{p\}}{\{p\}\ \text{while } B \text{ do } S \text{ od}\ \{p \wedge \neg B\}}$$

- Suppose now that for some assertion p

$$\mathcal{M}[[S]]([[p \wedge B]]) \subseteq [[p]]. \qquad (3.16)$$

- We prove by induction that **for all $k \geq 0$**

$$\mathcal{M}[[(\text{while } B \text{ do } S \text{ od})^k]]([[p]]) \subseteq [[p \wedge \neg B]].$$

- The case $k = 0$ is clear.

$$\boxed{\mathcal{M}[[\Omega]](\sigma) = \emptyset}$$

Lemma 3.3. (Input/Output)

(v) $\mathcal{M}[\![\text{while } B \text{ do } S \text{ od}]\!] = \bigcup_{k=0}^{\infty} \mathcal{M}[\![(\text{while } B \text{ do } S \text{ od})^k]\!]$.

ECNU

SEI

LOOP

RULE 5: LOOP

$$\frac{\{p \wedge B\}\ S\ \{p\}}{\{p\}\ \textbf{while}\ B\ \textbf{do}\ S\ \textbf{od}\ \{p \wedge \neg B\}}$$

$M[[S]]([[p \wedge B]]) \subseteq [[p]]$ (3.16)

- Suppose the claim holds for some $k > 0$. Then

$M[[(\textbf{while}\ B\ \textbf{do}\ S\ \textbf{od})^{k+1}]]([[p]])$

$=$ {definition of $(\textbf{while}\ B\ \textbf{do}\ S\ \textbf{od})^{k+1}$}

$M[[\textbf{if}\ B\ \textbf{then}\ S;\ (\textbf{while}\ B\ \textbf{do}\ S\ \textbf{od})^{k}\ \textbf{else}\ skip\ \textbf{fi}]]([[p]])$

$=$ {Input/Output Lemma 3.3(iv)}

$M[[S;\ (\textbf{while}\ B\ \textbf{do}\ S\ \textbf{od})^{k}]]([[p \wedge B]]) \cup M[[skip]]([[p \wedge \neg B]])$

$=$ {Input/Output Lemma 3.3(ii) and semantics of *skip*}

$M[[(\textbf{while}\ B\ \textbf{do}\ S\ \textbf{od})^{k}]](M[[S]]([[p \wedge B]])) \cup [[p \wedge \neg B]]$

$\subseteq$ {(3.16) and monotonicity of $M[[(\textbf{while}\ B\ \textbf{do}\ S\ \textbf{od})^{k}]]$}

$M[[(\textbf{while}\ B\ \textbf{do}\ S\ \textbf{od})^{k}]]([[p]]) \cup [[p \wedge \neg B]]$

$\subseteq$ {**induction hypothesis**}

$[[p \wedge \neg B]]$.

- This proves the induction step.

Lemma 3.3. (Input/Output)

(ii) $\mathcal{N}[\![S_1;\ S_2]\!](X) = \mathcal{N}[\![S_2]\!](\mathcal{N}[\![S_1]\!](X))$.

(iv) $\mathcal{N}[\![\textbf{if}\ B\ \textbf{then}\ S_1\ \textbf{else}\ S_2\ \textbf{fi}]\!](X) =$
$\mathcal{N}[\![S_1]\!](X \cap [\![B]\!]) \cup \mathcal{N}[\![S_2]\!](X \cap [\![\neg B]\!]) \cup \{\perp \mid \perp \in X\ \textit{and}\ \mathcal{N} = \mathcal{M}_{tot}\}$.

$\mathcal{M}[\![S]\!]([\![p \wedge B]\!]) \subseteq [\![p]\!]$. (3.16)

ECNU

SEI

LOOP

RULE 5: LOOP

$$\frac{\{p \wedge B\}\ S\ \{p\}}{\{p\}\ \text{while } B \text{ do } S \text{ od } \{p \wedge \neg B\}}$$

- Thus

$$\bigcup_{k=0}^{\infty} M[[(\textbf{while } B \textbf{ do } S \textbf{ od})^k]]([[p]]) \subseteq [[p \wedge \neg B]].$$

- But by the Input/Output Lemma 3.3(v)

$$M[[\textbf{while } B \textbf{ do } S \textbf{ od}]] = \bigcup_{k=0}^{\infty} M[[(\textbf{while } B \textbf{ do } S \textbf{ od})^k]];$$

- so

$$M[[\textbf{while } B \textbf{ do } S \textbf{ od}]]([[p]]) \subseteq [[p \wedge \neg B]].$$

- Thus the loop rule is sound for partial correctness.

Lemma 3.3. (Input/Output)

(v) $\mathcal{M}[\![\textbf{while } B \textbf{ do } S \textbf{ od}]\!] = \bigcup_{k=0}^{\infty} \mathcal{M}[\![(\textbf{while } B \textbf{ do } S \textbf{ od})^k]\!]$

ECNU

SEI

CONSEQUENCE

RULE 6: CONSEQUENCE

$$\frac{p \to p_1,\ \{p_1\}\ S\ \{q_1\},\ q_1 \to q}{\{p\}\ S\ \{q\}}$$

- Suppose that

  $p \to p_1$, $N[[S]]([[p_1]]) \subseteq [[q_1]]$, and $q_1 \to q$.

- Then, by the Meaning of Assertion Lemma 2.1, the inclusions $[[p]] \subseteq [[p_1]]$ and $[[q_1]] \subseteq [[q]]$ hold; so by the monotonicity of $N[[S]]$,

  $N[[S]]([[p]]) \subseteq N[[S]]([[p_1]]) \subseteq [[q_1]] \subseteq [[q]]$.

- Thus the consequence rule is sound for partial (total) correctness.

Lemma 2.1. (Meaning of Assertion)

(i) $[\![\neg p]\!] = \Sigma - [\![p]\!]$,
(ii) $[\![p \vee q]\!] = [\![p]\!] \cup [\![q]\!]$,
(iii) $[\![p \wedge q]\!] = [\![p]\!] \cap [\![q]\!]$,
(iv) $p \to q$ *is true iff* $[\![p]\!] \subseteq [\![q]\!]$,
(v) $p \leftrightarrow q$ *is true iff* $[\![p]\!] = [\![q]\!]$.

ECNU

SEI

- Thank you!