# A Denotational Engineering of Programming Languages

…

Part 7: Semantic correctness of programs
(Sections 7.1 – 7.6 of the book)

Andrzej Jacek Blikle

March 30th, 2020

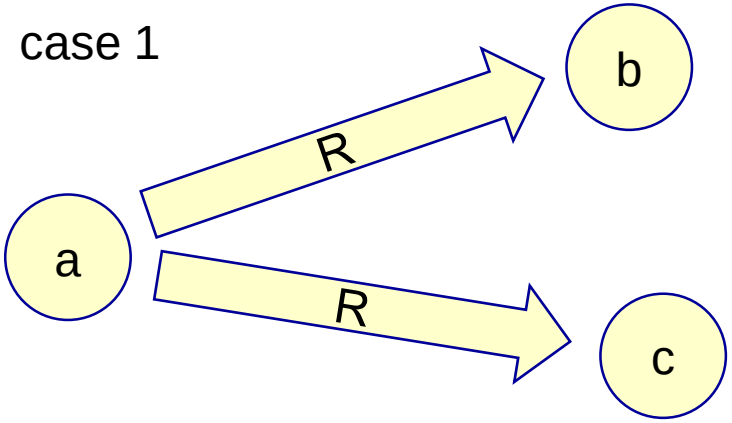
# Relational model of nondeterministic programs

S – a set of states

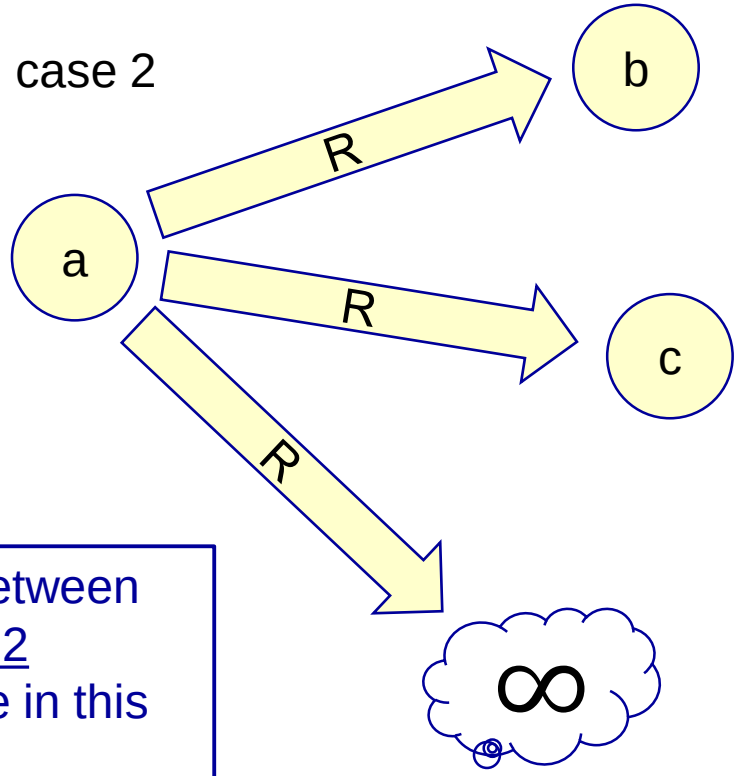
$R \subseteq S \times S$ – the denotation of a nondeterministic program

a R b – there is a finite (terminating) computation from a to b

case 1




case 2




In both cases
a R b and a R c

The difference between
case 1 and case 2
is not expressible in this model.

There are models for denotational treatment of Infinite executions.

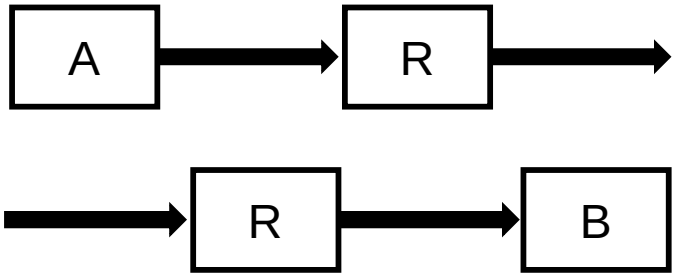
F.a = ? means either abortion or infinite run

# Composition of a relations with a set

Let R : Rel(S,S) and A, B ⊆S

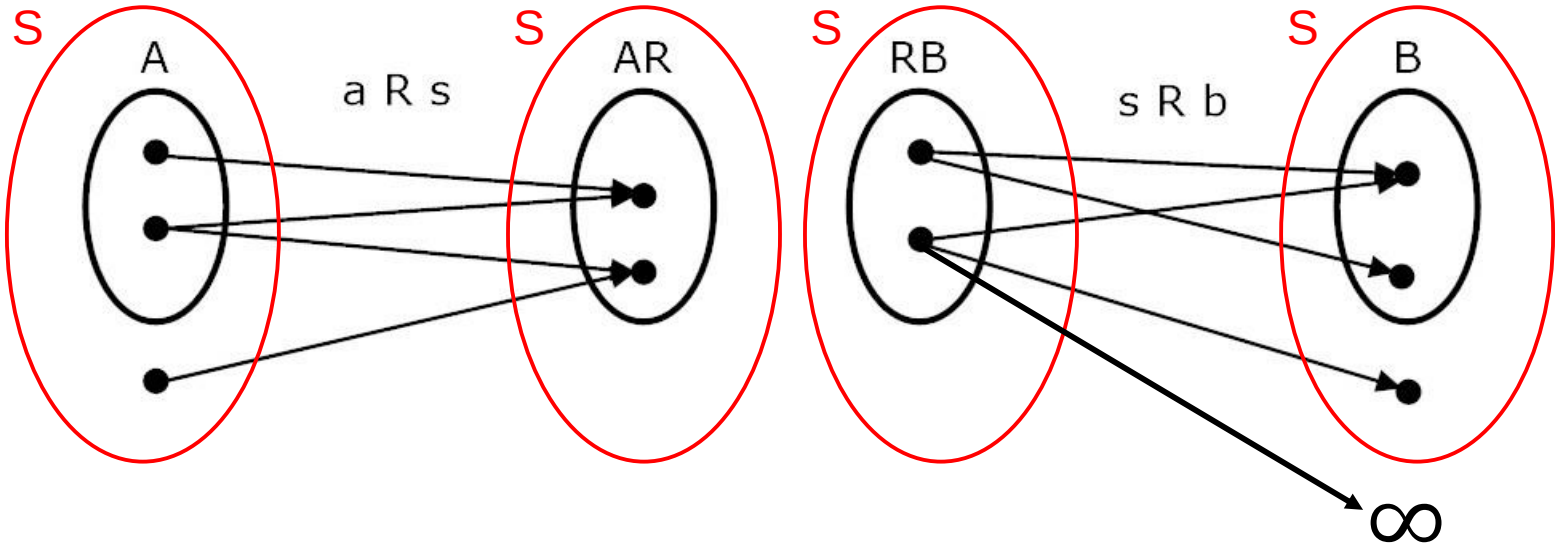
A R = {s | (∃a:A) a R s} – left composition; the image of A by R

R B = {s | (∃b:B) s R b} – right composition; the coimage of B by R.




AR final states of R-executions that start in A

RB initial states of R-executions that end in B




**A**

# Some properties of AR and RB

$A(RQ) = (AR)Q$ – associativity

$(RQ)B = R(QB)$

$(A \mid B)\, R = (AR) \mid (BR)$ – distributivity

$A\,(R \mid Q) = (AR) \mid (AQ)$

if $A \subseteq B$ then $AR \subseteq BR$ – monotonicity

if $R \subseteq Q$ then $AR \subseteq AQ$

$(\cup A_i)\, R = \cup\, (A_i\, R)$ – continuity

$A\,(\cup R_i) = \cup\, (A\, R_i)$

$R\,(\cup B_i) = \cup\, (R\, B_i)$ – continuity

$(\cup R_i)\, B = \cup\, (R_i\, B)$

# Structured programs in a relational framework

[A] : Rel(S,S) – an identity relation (function); [A] = {(a, a) | a : A}

3-valued partial predicates p on S will be represented by two disjoint sets of states

C = {s | p.s = tt}, C ∩ ¬C = Ø

¬C = {s | p.s = ff} C | ¬C ⊆S

S – (C | ¬C) – the set of states that lead to abortion or infinite executions

To distinguish between abortion and infinite execution we would need a third set:

eC = {s | p.s : Error}

We shall not exploit this option since in the construction of correct programs we avoid both – abortion and looping.

Constructors of structured programs:

P ; Q = P Q

**if** (C,¬C) **then** P **else** Q **fi** = [C] P | [¬C] Q

**while** (C,¬C) **do** P **od** = ([C]P)*[¬C]

i.e. the least solution of X = [C](PX) | [¬C]

A

# Program correctness

## general case – possibly nondeterministic

$AR \subseteq B$ – partial correctness wrt precondition A and postcondition B

$(\forall a : A)$ **if** $(\exists b)$ a R b **then** b : B

For every a:A, **every** a-execution of R which terminates, terminates in B.

$A \subseteq RB$ – weak total correctness wrt precondition A and postcondition B

$(\forall a : A)\ (\exists b)$ a R b **and** b : B

For every a:A, **there is** a-execution of R that terminates in B but there may be another executions, that do not terminate in B or do not terminate at all.

None of these properties is stronger than the other!

**A**

# Program correctness in deterministic case

deterministic case – F is a function

$AF \subseteq B$ – partial correctness: for every a : A, if F.a = ! then F.a : B

$A \subseteq FB$ – clean total correctness: for every a : A, F.a = ! and F.a : B

No abortion

$$A \subseteq FB \quad \text{iff} \quad AF \subseteq B \text{ and } F : A \mapsto S$$

Partial correctness

Clean termination in A

Clean termination = non-abortion & halting (no infinite execution)

Not a property of a function unless we have abstract errors (as in **Lingua**).

halting property
własność stopu

# Halting property of deterministic programs

In the general case halting property of programs is not decidable, and sometimes may be very difficult to prove.

```
pre n > 0
  x := n;
  while x > 1 do;
  if even(x) then x := x/2 else x := 3x + 1 fi
post x = 1
```

Collatz hypothesis formulated in 1937.

So far proved only for $n < 5*2^{86}$.

years > age of universe x $10^{65}$ with 1 ns cycle

```
pre n, m > 0
  x := 1; y := m;
  while x < n do;
  x := x+1;  y := y*m
post y = m^n
```

In some practical situations halting property may be quite obvious.

This makes us interested in partial correctness

# Proof rules for partial correctness

## No recursion or iteration

### Sequential composition

$$\frac{AP \subseteq B \quad B \subseteq C \quad CQ \subseteq D}{A(P;Q) \subseteq D}$$

### Strengthening precondition

$$\frac{AP \subseteq B \quad C \subseteq A}{CP \subseteq B}$$

### Conditional composition; $C \cap \neg C = \emptyset$

$$\frac{(A \cap C)P \subseteq B \quad (A \cap \neg C)Q \subseteq B}{A\ \textbf{if}\ (C,\neg C)\ \textbf{then}\ P\ \textbf{else}\ Q\ \textbf{fi} \subseteq B}$$

### Weakening postcondition

$$\frac{AP \subseteq B \quad B \subseteq C}{AP \subseteq C}$$

# The general case of (mutually) recursive procedures

$X_1 = \Psi_1.(X_1,\ldots,X_n)$
…
$X_n = \Psi_n.(X_1,\ldots,X_n)$

$\Psi_i$ – polynomials, e.g.

$\Psi.(X,Y,Z) = P\ X\ Q\ Y \mid X\ Y \mid P\ Z\ P$

There is nothing like canonical equations for recursion.
Each case has to be considered (given a rule) separately

Simple recursion

$X = HXT \mid E$

H – head
T – tail
E - exit

while is a particular case of simple recursion

$X = [C]PX \mid [\neg C]$
where
$H = [C]P$ , $T = [S]$ , $E = [\neg C]$

# Proof rules for partial correctness
## General recursion

A componentwise CPO of vectors of relations

$$\mathbf{R} = (R_1,\ldots,R_n) \qquad \mathbf{A} = (A_1,\ldots,A_n) \qquad n \geq 1$$

Let **R** be the least solution of **X = Ψ.X**,

General recursion

there exists a family of preconditions $\{\mathbf{A}_i \mid i \geq 0\}$
and a family of postconditions $\{\mathbf{B}_i \mid i \geq 0\}$ such that

$$\frac{(\forall i \geq 0)\ \mathbf{A}_i\ \boldsymbol{\Psi}^i.\boldsymbol{\emptyset} \subseteq \mathbf{B}_i \quad (\forall i \geq 0)\ \mathbf{A} \subseteq \mathbf{A}_i \quad \mathbf{U}\{\mathbf{B}_i \mid i \geq 0\} \subseteq \mathbf{B}}{\mathbf{A}\ \mathbf{R} \subseteq \mathbf{B}}$$

# Construction rules for partial correctness
## simple recursion

If R is the least solution of $X = HXT \mid E$ then for any $A, B \subseteq S$ the following rules hold:

Version 1

there exists a family of preconditions $\{A_i \mid i \geq 0\}$
and a family of postconditions $\{B_i \mid i \geq 0\}$ such that

$(\forall i \geq 0)\ A_i H^i E T^i \subseteq B_i$
$(\forall i \geq 0)\ A \subseteq A_i$
$U\{B_i \mid i \geq 0\} \subseteq B$

---

$AR \subseteq B$

Version 2 For any $A, B \subseteq S$

$(\forall Q)\ (AQ \subseteq B \text{ implies } A(HQT) \subseteq B)$
$AE \subseteq B$

---

$AR \subseteq B$

# Construction rules for partial correctness
## while loop

Then for any A, B ⊆S, any disjoint C, ¬C ⊆S, and for any P ⊆Rel(S, S)

there exists a family of postconditions $\{B_i \mid i \geq 0\}$ such that

$$\frac{(\forall i \geq 0)\ A\ ([C]P)^i\ [\neg C] \subseteq B_i \qquad \cup\{B_i \mid i \geq 0\} \subseteq B}{A\ \textbf{while}\ (C,\neg C)\ \textbf{do}\ P\ \textbf{od} \subseteq B}$$

there exists N ⊆S (called *loop invariant*) such that

$$\frac{(N \cap C)\ P \subseteq N \qquad A \subseteq N \qquad N\ [\neg C] \subseteq B}{A\ \textbf{while}\ (C,\neg C)\ \textbf{do}\ P\ \textbf{od} \subseteq B}$$

to prove ⬆set

N = A([C]P)*


Thank you for
your attention