

**a**  
**Practical**  
**Theory**  
**of**  
**Programming**

**2026-4-10 edition**

**Eric C.R. Hehner**

Department of Computer Science  
University of Toronto  
Toronto ON M5S 2E4 Canada

The first edition of this book was published by  
Springer-Verlag Publishers, New York, 1993  
ISBN 0-387-94106-1 QA76.6.H428

The current edition is available free at  
[hehner.ca/aPToP](http://hehner.ca/aPToP)

An on-line course based on this book is at  
[hehner.ca/FMSD](http://hehner.ca/FMSD)

The author's website is  
[hehner.ca](http://hehner.ca)

You may copy all or part of this book freely as long as you include this page.

The cover picture is an inukshuk, which is a human-like figure made of piled stones. Inukshuks are found throughout arctic Canada. They are built by the Inuit people, who use them to mean “You are on the right path.”.

## 11.3 Laws

### 11.3.0 Generic

The operators  $= \neq$  **if then else fi** apply to every type of expression (but the first operand of **if then else fi** must be binary), with the laws

$x=x$	Reflexivity	<b>if</b> $\top$ <b>then</b> $x$ <b>else</b> $y$ <b>fi</b> $= x$	Case Base
$x=y = y=x$	Symmetry	<b>if</b> $\perp$ <b>then</b> $x$ <b>else</b> $y$ <b>fi</b> $= y$	Case Base
$x=y \wedge y=z \Rightarrow x=z$	Transitivity	<b>if</b> $a$ <b>then</b> $x$ <b>else</b> $x$ <b>fi</b> $= x$	Case Idempotent
$x=y \Rightarrow f.x = f.y$	Transparency	<b>if</b> $a$ <b>then</b> $x$ <b>else</b> $y$ <b>fi</b> $=$ <b>if</b> $\neg a$ <b>then</b> $y$ <b>else</b> $x$ <b>fi</b>	Case Reversal
$x \neq y = \neg(x=y)$	Unequality		

The operators  $\uparrow \downarrow < \leq > \geq$  apply to numbers, characters, strings, and lists, with the laws

$x \leq y = x = x \downarrow y$		$x \downarrow y \leq x \leq x \uparrow y$		$x \leq y = y = x \uparrow y$
$x \leq x$	Reflexivity	$\neg x < x$	Irreflexivity	
$\neg(x < y \wedge x = y)$	Exclusivity	$\neg(x > y \wedge x = y)$	Exclusivity	
$\neg(x < y \wedge x > y)$	Exclusivity	$x \leq y = x < y \vee x = y$	Inclusivity	
$x \leq y \wedge y \leq z \Rightarrow x \leq z$	Transitivity	$x < y \wedge y < z \Rightarrow x < z$	Transitivity	
$x < y \wedge y < z \Rightarrow x < z$	Transitivity	$x \leq y \wedge y < z \Rightarrow x < z$	Transitivity	
$x > y = y < x$	Mirror	$x \geq y = y \leq x$	Mirror	
$\neg x < y = x \geq y$	Totality	$\neg x \leq y = x > y$	Totality	
$x \leq y \wedge y \leq x = x = y$	Antisymmetry	$x < y \vee x = y \vee x > y$	Totality, Trichotomy	
$x \uparrow x = x$	Idempotence	$x \downarrow x = x$	Idempotence	
$x \uparrow y = y \uparrow x$	Symmetry	$x \downarrow y = y \downarrow x$	Symmetry	
$x \uparrow (y \uparrow z) = (x \uparrow y) \uparrow z$	Associativity	$x \downarrow (y \downarrow z) = (x \downarrow y) \downarrow z$	Associativity	
$x \uparrow (y \downarrow z) = (x \uparrow y) \downarrow (x \uparrow z)$	Distributivity	$x \downarrow (y \uparrow z) = (x \downarrow y) \uparrow (x \downarrow z)$	Distributivity	
$x \uparrow y \leq z = x \leq z \wedge y \leq z$	Connection	$x \downarrow y \leq z = x \leq z \vee y \leq z$	Connection	
$x \leq y \uparrow z = x \leq y \vee x \leq z$	Connection	$x \leq y \downarrow z = x \leq y \wedge x \leq z$	Connection	
$x \uparrow y =$ <b>if</b> $x \geq y$ <b>then</b> $x$ <b>else</b> $y$ <b>fi</b>		$x \downarrow y =$ <b>if</b> $x \leq y$ <b>then</b> $x$ <b>else</b> $y$ <b>fi</b>		

—End of Generic

### 11.3.1 Binary

Let  $a, b, c, d,$  and  $e$  be binary.

Binary

$\top$   
 $\neg \perp$   
 $\top \neq \perp$

Excluded Middle

$a \vee \neg a$

Noncontradiction

$\neg(a \wedge \neg a)$

Base

$\neg(a \wedge \perp)$   
 $a \vee \top$   
 $a \Rightarrow \top$   
 $\perp \Rightarrow a$

Mirror

$a \Leftarrow b = b \Rightarrow a$

Double Negation

$\neg \neg a = a$

Duality

$\neg(a \wedge b) = \neg a \vee \neg b$

$\neg(a \vee b) = \neg a \wedge \neg b$

Exclusion

$a \Rightarrow \neg b = b \Rightarrow \neg a$  (Contrapositive)

$a = \neg b = a \neq b = \neg a = b$

Inclusion

$a \Rightarrow b = \neg a \vee b$  (Material Implication)

$a \Rightarrow b = (a \wedge b = a)$

$a \Rightarrow b = (a \vee b = b)$

## Identity

$$\begin{aligned}\top \wedge a &= a \\ \perp \vee a &= a \\ \top \Rightarrow a &= a \\ \top = a &= a\end{aligned}$$

## Idempotent

$$\begin{aligned}a \wedge a &= a \\ a \vee a &= a\end{aligned}$$

## Reflexive

$$\begin{aligned}a \Rightarrow a \\ a = a\end{aligned}$$

## Indirect Proof

$$\begin{aligned}\neg a \Rightarrow \perp &= a \\ \neg a \Rightarrow a &= a\end{aligned}$$

## Specialization

$$a \wedge b \Rightarrow a$$

## Associative

$$\begin{aligned}a \wedge (b \wedge c) &= (a \wedge b) \wedge c \\ a \vee (b \vee c) &= (a \vee b) \vee c \\ a = (b = c) &= (a = b) = c \\ a \neq (b \neq c) &= (a \neq b) \neq c \\ a = (b \neq c) &= (a = b) \neq c\end{aligned}$$

## Symmetry (Commutative)

$$\begin{aligned}a \wedge b &= b \wedge a \\ a \vee b &= b \vee a \\ a = b &= b = a \\ a \neq b &= b \neq a\end{aligned}$$

## Antisymmetry (Double Implication)

$$(a \Rightarrow b) \wedge (b \Rightarrow a) = a = b$$

## Discharge

$$\begin{aligned}a \wedge (a \Rightarrow b) &= a \wedge b \\ a \Rightarrow (a \wedge b) &= a \Rightarrow b\end{aligned}$$

## Antimonotonic

$$\begin{aligned}a \Rightarrow b &= \neg a \Leftarrow \neg b \text{ (Contrapositive)} \\ a \Rightarrow b &\Rightarrow (a \Rightarrow c) \Leftarrow (b \Rightarrow c)\end{aligned}$$

## Monotonic

$$\begin{aligned}a \Rightarrow b &\Rightarrow a \wedge c \Rightarrow b \wedge c \\ a \Rightarrow b &\Rightarrow a \vee c \Rightarrow b \vee c \\ a \Rightarrow b &\Rightarrow (c \Rightarrow a) \Rightarrow (c \Rightarrow b)\end{aligned}$$

## Absorption

$$\begin{aligned}a \wedge (a \vee b) &= a \\ a \vee (a \wedge b) &= a\end{aligned}$$

## Direct Proof

$$\begin{aligned}(a \Rightarrow b) \wedge a &\Rightarrow b \\ (a \Rightarrow b) \wedge \neg b &\Rightarrow \neg a \\ (a \vee b) \wedge \neg a &\Rightarrow b\end{aligned}$$

## Transitive

$$\begin{aligned}(a \wedge b) \wedge (b \wedge c) &\Rightarrow (a \wedge c) \\ (a \Rightarrow b) \wedge (b \Rightarrow c) &\Rightarrow (a \Rightarrow c) \\ (a = b) \wedge (b = c) &\Rightarrow (a = c) \\ (a \Rightarrow b) \wedge (b = c) &\Rightarrow (a \Rightarrow c) \\ (a = b) \wedge (b \Rightarrow c) &\Rightarrow (a \Rightarrow c)\end{aligned}$$

## Distributive (Factoring)

$$\begin{aligned}a \wedge (b \wedge c) &= (a \wedge b) \wedge (a \wedge c) \\ a \wedge (b \vee c) &= (a \wedge b) \vee (a \wedge c) \\ a \vee (b \wedge c) &= (a \vee b) \wedge (a \vee c) \\ a \vee (b \vee c) &= (a \vee b) \vee (a \vee c) \\ a \vee (b \Rightarrow c) &= (a \vee b) \Rightarrow (a \vee c) \\ a \vee (b = c) &= (a \vee b) = (a \vee c) \\ a \Rightarrow (b \wedge c) &= (a \Rightarrow b) \wedge (a \Rightarrow c) \\ a \Rightarrow (b \vee c) &= (a \Rightarrow b) \vee (a \Rightarrow c) \\ a \Rightarrow (b \Rightarrow c) &= (a \Rightarrow b) \Rightarrow (a \Rightarrow c) \\ a \Rightarrow (b = c) &= (a \Rightarrow b) = (a \Rightarrow c)\end{aligned}$$

## Generalization

$$a \Rightarrow a \vee b$$

## Antidistributive

$$\begin{aligned}a \wedge b \Rightarrow c &= (a \Rightarrow c) \vee (b \Rightarrow c) \\ a \vee b \Rightarrow c &= (a \Rightarrow c) \wedge (b \Rightarrow c)\end{aligned}$$

## Portation

$$\begin{aligned}a \wedge b \Rightarrow c &= a \Rightarrow (b \Rightarrow c) \\ a \wedge b \Rightarrow c &= a \Rightarrow \neg b \vee c\end{aligned}$$

## Conflation

$$\begin{aligned}(a \Rightarrow b) \wedge (c \Rightarrow d) &\Rightarrow a \wedge c \Rightarrow b \wedge d \\ (a \Rightarrow b) \wedge (c \Rightarrow d) &\Rightarrow a \vee c \Rightarrow b \vee d\end{aligned}$$

## Equality and Difference

$$\begin{aligned}a = b &= (a \wedge b) \vee (\neg a \wedge \neg b) \\ a \neq b &= (a \wedge \neg b) \vee (\neg a \wedge b)\end{aligned}$$

Resolution  $a \wedge c \implies (a \vee b) \wedge (\neg b \vee c) \equiv (a \wedge \neg b) \vee (b \wedge c) \implies a \vee c$

Case Creation

$a = \mathbf{if\ } b \mathbf{\ then\ } b \implies a \mathbf{\ else\ } \neg b \implies a \mathbf{\ fi}$   
 $a = \mathbf{if\ } b \mathbf{\ then\ } b \wedge a \mathbf{\ else\ } \neg b \wedge a \mathbf{\ fi}$   
 $a = \mathbf{if\ } b \mathbf{\ then\ } b = a \mathbf{\ else\ } b \neq a \mathbf{\ fi}$

Case Analysis

$\mathbf{if\ } a \mathbf{\ then\ } b \mathbf{\ else\ } c \mathbf{\ fi} \equiv (a \wedge b) \vee (\neg a \wedge c)$   
 $\mathbf{if\ } a \mathbf{\ then\ } b \mathbf{\ else\ } c \mathbf{\ fi} \equiv (a \implies b) \wedge (\neg a \implies c)$

Case Absorption

$\mathbf{if\ } a \mathbf{\ then\ } b \mathbf{\ else\ } c \mathbf{\ fi} = \mathbf{if\ } a \mathbf{\ then\ } a \wedge b \mathbf{\ else\ } c \mathbf{\ fi}$   
 $\mathbf{if\ } a \mathbf{\ then\ } b \mathbf{\ else\ } c \mathbf{\ fi} = \mathbf{if\ } a \mathbf{\ then\ } a \implies b \mathbf{\ else\ } c \mathbf{\ fi}$   
 $\mathbf{if\ } a \mathbf{\ then\ } b \mathbf{\ else\ } c \mathbf{\ fi} = \mathbf{if\ } a \mathbf{\ then\ } a = b \mathbf{\ else\ } c \mathbf{\ fi}$   
 $\mathbf{if\ } a \mathbf{\ then\ } b \mathbf{\ else\ } c \mathbf{\ fi} = \mathbf{if\ } a \mathbf{\ then\ } b \mathbf{\ else\ } \neg a \wedge c \mathbf{\ fi}$   
 $\mathbf{if\ } a \mathbf{\ then\ } b \mathbf{\ else\ } c \mathbf{\ fi} = \mathbf{if\ } a \mathbf{\ then\ } b \mathbf{\ else\ } a \vee c \mathbf{\ fi}$   
 $\mathbf{if\ } a \mathbf{\ then\ } b \mathbf{\ else\ } c \mathbf{\ fi} = \mathbf{if\ } a \mathbf{\ then\ } b \mathbf{\ else\ } a \neq c \mathbf{\ fi}$

One Case

$\mathbf{if\ } a \mathbf{\ then\ } \top \mathbf{\ else\ } b \mathbf{\ fi} = a \vee b$   
 $\mathbf{if\ } a \mathbf{\ then\ } \perp \mathbf{\ else\ } b \mathbf{\ fi} = \neg a \wedge b$   
 $\mathbf{if\ } a \mathbf{\ then\ } b \mathbf{\ else\ } \top \mathbf{\ fi} = a \implies b$   
 $\mathbf{if\ } a \mathbf{\ then\ } b \mathbf{\ else\ } \perp \mathbf{\ fi} = a \wedge b$   
 $\mathbf{if\ } a \mathbf{\ then\ } b \mathbf{\ else\ } \neg b \mathbf{\ fi} = a = b$   
 $\mathbf{if\ } a \mathbf{\ then\ } \neg b \mathbf{\ else\ } b \mathbf{\ fi} = a \neq b$

Case Distributive (Case Factoring)

$\neg \mathbf{if\ } a \mathbf{\ then\ } b \mathbf{\ else\ } c \mathbf{\ fi} = \mathbf{if\ } a \mathbf{\ then\ } \neg b \mathbf{\ else\ } \neg c \mathbf{\ fi}$   
 $\mathbf{if\ } a \mathbf{\ then\ } b \mathbf{\ else\ } c \mathbf{\ fi} \wedge d = \mathbf{if\ } a \mathbf{\ then\ } b \wedge d \mathbf{\ else\ } c \wedge d \mathbf{\ fi}$   
 and similarly replacing  $\wedge$  by any of  $\vee = \neq \implies \Leftarrow$   
 $\mathbf{if\ } a \mathbf{\ then\ } b \wedge c \mathbf{\ else\ } d \wedge e \mathbf{\ fi} = \mathbf{if\ } a \mathbf{\ then\ } b \mathbf{\ else\ } d \mathbf{\ fi} \wedge \mathbf{if\ } a \mathbf{\ then\ } c \mathbf{\ else\ } e \mathbf{\ fi}$   
 and similarly replacing  $\wedge$  by any of  $\vee = \neq \implies \Leftarrow$

End of Binary

### 11.3.2 Numbers

Let  $d$  be a sequence of (zero or more) digits, and let  $x$ ,  $y$ , and  $z$  be numbers.

$d0+1 = d1$	$d5+1 = d6$	Counting (see Exercise 32)
$d1+1 = d2$	$d6+1 = d7$	Counting
$d2+1 = d3$	$d7+1 = d8$	Counting
$d3+1 = d4$	$d8+1 = d9$	Counting
$d4+1 = d5$	$9+1 = 10$	Counting
for nonempty $d$	$d9+1 = (d+1)0$	Counting
$x+0 = x$		Identity
$x+y = y+x$		Symmetry
$x+(y+z) = (x+y)+z$		Associativity
$-\infty < x < \infty \implies (x+y = x+z \equiv y=z)$		Cancellation
$-\infty < x \implies \infty + x = \infty$		Absorption
$x < \infty \implies -\infty + x = -\infty$		Absorption
$-x = 0-x$		Negation
$--x = x$		Self-inverse
$-(x+y) = -x + -y$		Distributivity
$-(x-y) = y-x$		Antisymmetry
$-x \times y = -(x \times y) = x \times -y$		Semi-distributivity
$-x / y = -(x/y) = x / -y$		Semi-distributivity
$x-0 = x$		Identity
$x-y = x + -y$		Subtraction
$x+(y-z) = (x+y)-z$		Addition-Subtraction
$x-(y+z) = (x-y)-z$		Addition-Subtraction
$-\infty < x < \infty \implies (x-y = x-z \equiv y=z)$		Cancellation
$-\infty < x < \infty \implies x-x = 0$		Inverse
$x < \infty \implies \infty - x = \infty$		Absorption
$-\infty < x \implies -\infty - x = -\infty$		Absorption
$-\infty < x < \infty \implies x \times 0 = 0$		Base
$x \times 1 = x$		Identity

$x \times y = y \times x$	Symmetry
$x \times (y+z) = x \times y + x \times z$	Distributivity
$x \times (y \times z) = (x \times y) \times z$	Associativity
$-\infty < x < \infty \wedge x \neq 0 \Rightarrow (x \times y = x \times z \Rightarrow y = z)$	Cancellation
$0 < x \Rightarrow x \times \infty = \infty$	Absorption
$0 < x \Rightarrow x \times -\infty = -\infty$	Absorption
$x/1 = x$	Identity
$x \neq 0 \Rightarrow 0/x = 0$	Base
$-\infty < x < \infty \wedge x \neq 0 \Rightarrow x/x = 1$	Base
$x \times (y/z) = (x \times y)/z = (x/z) \times y = x/(z/y)$	Multiplication-Division
$(x/y)/z = x/(y \times z)$	Multiplication-Division
$-\infty < y < \infty \wedge y \neq 0 \Rightarrow (x/y) \times y = x$	Multiplication-Division
$-\infty < x < \infty \Rightarrow x/\infty = 0 = x/-\infty$	Annihilation
$-\infty < x < \infty \Rightarrow x^0 = 1$	Base
$x^1 = x$	Identity
$-\infty < 0 < 1 < \infty$	Direction
$x < y \Rightarrow -y < -x$	Reflection
$-\infty < x < \infty \Rightarrow (x+y < x+z \Rightarrow y < z)$	Cancellation, Translation
$0 < x < \infty \Rightarrow (x \times y < x \times z \Rightarrow y < z)$	Cancellation, Scale
$x < y \vee x = y \vee x > y$	Trichotomy
$-\infty \leq x \leq \infty$	Extremes
$x \uparrow \infty = \infty$	Base
$x \uparrow -\infty = x$	Identity
$-(x \uparrow y) = -x \downarrow -y$	Duality
$x + y \uparrow z = (x+y) \uparrow (x+z)$	Distributivity
$x \geq 0 \Rightarrow x \times (y \uparrow z) = (x \times y) \uparrow (x \times z)$	Distributivity
$x \leq 0 \Rightarrow x \times (y \uparrow z) = (x \times y) \downarrow (x \times z)$	Distributivity
$x \downarrow -\infty = -\infty$	Base
$x \downarrow \infty = x$	Identity
$-(x \downarrow y) = -x \uparrow -y$	Duality
$x - y \uparrow z = (x-y) \downarrow (x-z)$	Distributivity
$x \geq 0 \Rightarrow x \times (y \downarrow z) = (x \times y) \downarrow (x \times z)$	Distributivity
$x \leq 0 \Rightarrow x \times (y \downarrow z) = (x \times y) \uparrow (x \times z)$	Distributivity

End of Numbers

### 11.3.3 Bunches

Let  $x$  and  $y$  be elements (binaries, numbers, characters, sets, strings and lists of elements).

$x: y = x=y$	Elementary
$x: A, B = x: A \vee x: B$	Union
$x: A \text{ ' } B = x: A \wedge x: B$	Intersection
$x: A \bar{\text{ ' }} B = x: A \wedge \neg x: B$	Removal
$A, A = A$	Idempotence
$A, B = B, A$	Symmetry
$A, (B, C) = (A, B), C$	Associativity
$A \text{ ' } A = A$	Idempotence
$A \text{ ' } B = B \text{ ' } A$	Symmetry
$A \text{ ' } (B \text{ ' } C) = (A \text{ ' } B) \text{ ' } C$	Associativity
$A, B: C = A: C \wedge B: C$	Antidistributivity
$A: B \text{ ' } C = A: B \wedge A: C$	Distributivity
$A: A, B$	Generalization
$A \text{ ' } B: A$	Specialization
$A: A$	Reflexivity
$A: B \wedge B: A = A=B$	Antisymmetry
$A: B \wedge B: C \Rightarrow A: C$	Transitivity
$A:: B = B: A$	Mirror

$\emptyset \text{ null} = 0$	Size
$\emptyset x = 1$	Size
$\emptyset \text{ nat} = \infty$	Size
$\emptyset(A, B) + \emptyset(A'B) = \emptyset A + \emptyset B$	Size
$\neg x: A = \emptyset(A'x) = 0$	Size
$A: B \Rightarrow \emptyset A \leq \emptyset B$	Size
$A, (A'B) = A = A'(A, B)$	Absorption
$A: B = A, B = B = A = A'B$	Inclusion
$A, (B, C) = (A, B), (A, C)$	Distributivity
$A, (B'C) = (A, B)'(A, C)$	Distributivity
$A'(B, C) = (A'B), (A'C)$	Distributivity
$A'(B'C) = (A'B)'(A'C)$	Distributivity
$A_{\neg}(B, C) = (A_{\neg}B)_{\neg}C = (A_{\neg}B)'(A_{\neg}C)$	Union Removal
$(A, B)_{\neg}C = A_{\neg}C, B_{\neg}C$	Union Removal
$A'(B_{\neg}C) = (A'B)_{\neg}C = B'(A_{\neg}C)$	Intersection Removal
$A: B \wedge C: D \Rightarrow A, C: B, D$	Conflation, Monotonicity
$A: B \wedge C: D \Rightarrow A': C: B': D$	Conflation, Monotonicity
$\text{null}: A$	Induction
$A, \text{null} = A = \text{null}, A$	Identity
$A'\text{null} = \text{null} = \text{null}'A$	Base
$\emptyset A = 0 = A = \text{null}$	Size
$x, y: x \text{int} \wedge x \leq y \Rightarrow (i: x, ..y = i: \text{int} \wedge x \leq i < y)$	Interval
$x, y: x \text{int} \wedge x \leq y \Rightarrow \emptyset(x, ..y) = y - x$	Interval
$\text{nat} = 0, ..\infty$	Interval
$\infty, -\infty: x/0$	Division by 0
$x \text{real}: 0/0$	Division by 0
$x^y+z: x^y \times x^z$	Adding Exponents
$x^y \times z: (x^y)^z$	Multiplying Exponents
$-\text{null} = \text{null}$	Distribution
$-(A, B) = -A, -B$	Distribution
$A + \text{null} = \text{null} = \text{null} + A$	Distribution
$(A, B) + (C, D) = A + C, A + D, B + C, B + D$	Distribution

and similarly for many other operators (see Section [11.7](#))

End of Bunches

### 11.3.4 Sets

Let  $S$  be a set.

$\{\sim S\} = S$	$\{A\}: \not\{B\} = A: B$
$\sim\{A\} = A$	$\$\{A\} = \emptyset A$
$\{A\} \neq A$	$\{A\} \cup \{B\} = \{A, B\}$
$A \in \{B\} = A: B$	$\{A\} \cap \{B\} = \{A' B\}$
$\{A\} \subseteq \{B\} = A: B$	$\{A\} = \{B\} = A = B$
	$\{A\} \neq \{B\} = A \neq B$

End of Sets

### 11.3.5 Strings

Let  $S$ ,  $T$ , and  $U$  be strings; let  $i$  and  $j$  be items (binary values, numbers, characters, sets, lists, functions); let  $n$  and  $m$  be extended natural; let  $x$ ,  $y$ , and  $z$  be extended integers such that  $x \leq y \leq z$ .

$S; nil = S = nil; S$ $S; (T; U) = (S; T); U$ $\leftrightarrow nil = 0$ $\leftrightarrow i = 1$ $\leftrightarrow (S; T) = \leftrightarrow S + \leftrightarrow T$ $\not\leftrightarrow nil = 1$ $\not\leftrightarrow (A; B) \leq \not\leftrightarrow A \times \not\leftrightarrow B$ $\leftrightarrow S < \infty \Rightarrow (S; i; T) \leftrightarrow_S = i$ $\leftrightarrow S < \infty \Rightarrow S; i; T \triangleleft \leftrightarrow S \triangleright j = S; j; T$ $0 * S = nil$ $(n+1) * S = n * S; S$ $* S = ** S = nat * S$	$S_{(TU)} = (S_T)_U$ $S_{nil} = nil$ $S_{T; U} = S_T; S_U$ $S_{\{A\}} = \{S_A\}$ $\leftrightarrow S < \infty \Rightarrow nil \leq S < S; i; T$ $\leftrightarrow S < \infty \wedge i < j \Rightarrow S; i; T < S; j; U$ $\leftrightarrow S < \infty \Rightarrow (S; A; T; S; B; T = A; B)$ $\leftrightarrow S < \infty \Rightarrow (i=j = S; i; T = S; j; T)$ $(S \triangleleft n \triangleright i)_m = \mathbf{if} \ n=m \ \mathbf{then} \ i \ \mathbf{else} \ S_m \ \mathbf{fi}$ $-\infty < x < \infty \Rightarrow x; ..x = nil$ $-\infty < x < \infty \Rightarrow x; ..x+1 = x$ $(x; ..y); (y; ..z) = x; ..z$ $\leftrightarrow (x; ..y) = y-x$
--	--

End of Strings

### 11.3.6 Lists

Let  $S$  and  $T$  be strings; let  $i$  be an item (binary value, number, character, set, list, function); let  $L$ ,  $M$ , and  $N$  be lists; let  $n$  and  $m$  be extended natural.

$[S] \# S = \sim[S]$ $[\sim L] = L$ $[S];; [T] = [S; T]$ $[S] = [T] = S = T$ $[S] < [T] = S < T$ $[A]; [B] = A; B$ $\#[S] = \leftrightarrow S$ $nil \rightarrow i \mid L = i$ $n \rightarrow i \mid [S] = [S \triangleleft n \triangleright i]$ $(n \rightarrow i \mid L) m = \mathbf{if} \ n=m \ \mathbf{then} \ i \ \mathbf{else} \ L m \ \mathbf{fi}$ $(S; T) \rightarrow i \mid L = S \rightarrow (T \rightarrow i \mid L @ S) \mid L$	$\square L = 0, .. \#L$ $[S] T = S_T$ $S_{[T]} = [S_T]$ $[S] [T] = [S_T]$ $L \{A\} = \{L A\}$ $L [S] = [L S]$ $(L M) N = L (M N)$ $\#L = \not\leftrightarrow \square L$ $L @ nil = L$ $L @ i = L i$ $L @ (S; T) = L @ S @ T$
--	--

End of Lists

### 11.3.7 Functions

Renaming — if  $v$  and  $w$  do not appear in  $D$

$$\text{and } w \text{ does not appear in } b$$

$$\langle v: D \cdot b \rangle = \langle w: D \cdot \langle v: D \cdot b \rangle w \rangle$$

Function Union

$$\square (f, g) = \square f \cup \square g$$

$$(f, g) x = f x, g x$$

Function Composition — if  $\neg f: \square g$

$$\square (g f) = \S x: \square f f x: \square g$$

$$(g f) x = g (f x)$$

Function Intersection

$$\square (f \cap g) = \square f \cap \square g$$

$$(f \cap g) x = (f \mid g) x \cap (g \mid f) x$$

Domain

$$\square \langle v: D \cdot b \rangle = D$$

Selective Union

$$\square (f \mid g) = \square f \cup \square g$$

$$(f \mid g) x = \mathbf{if} \ x: \square f \ \mathbf{then} \ f x \ \mathbf{else} \ g x \ \mathbf{fi}$$

$$f \mid f = f$$

$$f \mid (g \mid h) = (f \mid g) \mid h$$

$$(g \mid h) f = g f \mid h f$$

Application — if element  $x: D$

$$\langle v: D \cdot b \rangle x = (\text{substitute } x \text{ for } v \text{ in } b)$$

Distributive

$$f \text{ null} = \text{null}$$

$$f(A, B) = f A, f B$$

$$f(\S g) = \S y: f(\square g) \cdot \exists x: \square g \cdot f x = y \wedge g x$$

$$f \ \mathbf{if} \ b \ \mathbf{then} \ x \ \mathbf{else} \ y \ \mathbf{fi} = \mathbf{if} \ b \ \mathbf{then} \ f x \ \mathbf{else} \ f y \ \mathbf{fi}$$

$$\mathbf{if} \ b \ \mathbf{then} \ f \ \mathbf{else} \ g \ \mathbf{fi} \ x = \mathbf{if} \ b \ \mathbf{then} \ f x \ \mathbf{else} \ g x \ \mathbf{fi}$$

Function Inclusion and Equality

$$f: g = \square f: \square g \wedge \forall x: \square g \cdot f x: g x$$

$$f = g = \square f = \square g \wedge \forall x: \square f \cdot f x = g x$$

$$f: A \rightarrow B = \square f: A \wedge \forall a: A \cdot f a: B$$

Arrow

$$\begin{aligned}
f: \text{null} \rightarrow A \\
A \rightarrow B : C \rightarrow D &= A::C \wedge B:D \\
(A, B) \rightarrow (C, D) : A \rightarrow C : (A, B) \rightarrow (C, D) \\
(A, B) \rightarrow C &= A \rightarrow C \mid B \rightarrow C = A \rightarrow C \text{ ' } B \rightarrow C
\end{aligned}$$

Size

$$\#f = \phi \square f$$

Extension

$$f = \langle v: \square f \cdot f v \rangle$$

End of Functions

### 11.3.8 Quantifiers

Let  $x$  be an element, let  $a$ ,  $b$  and  $c$  be binary, let  $n$  and  $m$  be numeric, let  $f$  and  $g$  be functions, and let  $p$  be a predicate.

$$\begin{aligned}
\forall v: \text{null} \cdot b &= \top & \forall v: A, B \cdot b &= (\forall v: A \cdot b) \wedge (\forall v: B \cdot b) \\
\forall v: x \cdot b &= \langle v: x \cdot b \rangle x & \forall v: (\S v: D \cdot b) \cdot c &= \forall v: D \cdot b \Rightarrow c
\end{aligned}$$

$$\begin{aligned}
\exists v: \text{null} \cdot b &= \perp & \exists v: A, B \cdot b &= (\exists v: A \cdot b) \vee (\exists v: B \cdot b) \\
\exists v: x \cdot b &= \langle v: x \cdot b \rangle x & \exists v: (\S v: D \cdot b) \cdot c &= \exists v: D \cdot b \wedge c
\end{aligned}$$

$$\begin{aligned}
\Sigma v: \text{null} \cdot n &= 0 & (\Sigma v: A, B \cdot n) + (\Sigma v: A' B \cdot n) &= (\Sigma v: A \cdot n) + (\Sigma v: B \cdot n) \\
\Sigma v: x \cdot n &= \langle v: x \cdot n \rangle x & \Sigma v: (\S v: D \cdot b) \cdot n &= \Sigma v: D \cdot \mathbf{if } b \mathbf{ then } n \mathbf{ else } 0 \mathbf{ fi}
\end{aligned}$$

$$\begin{aligned}
\Pi v: \text{null} \cdot n &= 1 & (\Pi v: A, B \cdot n) \times (\Pi v: A' B \cdot n) &= (\Pi v: A \cdot n) \times (\Pi v: B \cdot n) \\
\Pi v: x \cdot n &= \langle v: x \cdot n \rangle x & \Pi v: (\S v: D \cdot b) \cdot n &= \Pi v: D \cdot \mathbf{if } b \mathbf{ then } n \mathbf{ else } 1 \mathbf{ fi}
\end{aligned}$$

$$\begin{aligned}
\Downarrow v: \text{null} \cdot n &= \infty & \Downarrow v: A, B \cdot n &= (\Downarrow v: A \cdot n) \Downarrow (\Downarrow v: B \cdot n) \\
\Downarrow v: x \cdot n &= \langle v: x \cdot n \rangle x & \Downarrow v: (\S v: D \cdot b) \cdot n &= \Downarrow v: D \cdot \mathbf{if } b \mathbf{ then } n \mathbf{ else } \infty \mathbf{ fi}
\end{aligned}$$

$$\begin{aligned}
\Uparrow v: \text{null} \cdot n &= -\infty & \Uparrow v: A, B \cdot n &= (\Uparrow v: A \cdot n) \Uparrow (\Uparrow v: B \cdot n) \\
\Uparrow v: x \cdot n &= \langle v: x \cdot n \rangle x & \Uparrow v: (\S v: D \cdot b) \cdot n &= \Uparrow v: D \cdot \mathbf{if } b \mathbf{ then } n \mathbf{ else } -\infty \mathbf{ fi}
\end{aligned}$$

$$\begin{aligned}
\S v: \text{null} \cdot b &= \text{null} & \text{Inclusion} & \\
\S v: x \cdot b &= \mathbf{if } \langle v: x \cdot b \rangle x \mathbf{ then } x \mathbf{ else null fi} & A: B &= \forall x: A \cdot x: B \\
\S v: A, B \cdot b &= (\S v: A \cdot b), (\S v: B \cdot b) & \text{Cardinality} & \\
\S v: A' B \cdot b &= (\S v: A \cdot b) \text{ ' } (\S v: B \cdot b) & \phi A &= \Sigma (A \rightarrow 1) \\
\S v: (\S v: D \cdot b) \cdot c &= \S v: D \cdot b \wedge c & &
\end{aligned}$$

Change of Variable — if  $d$  does not appear in  $b$

$$\begin{aligned}
\forall r: f D \cdot b &= \forall d: D \cdot \langle r: f D \cdot b \rangle (f d) \\
\exists r: f D \cdot b &= \exists d: D \cdot \langle r: f D \cdot b \rangle (f d) \\
\Downarrow r: f D \cdot n &= \Downarrow d: D \cdot \langle r: f D \cdot n \rangle (f d) \\
\Uparrow r: f D \cdot n &= \Uparrow d: D \cdot \langle r: f D \cdot n \rangle (f d)
\end{aligned}$$

Identity

$$\begin{aligned}
\forall v \cdot \top \\
\neg \exists v \cdot \perp
\end{aligned}$$

Specialize and Generalize — if element  $x: \square f$

$$\Downarrow f \leq f x \leq \Uparrow f$$

Bunch-Element Conversion

$$\begin{aligned}
A: B &= \forall a: A \exists b: B \cdot a=b \\
f A: g B &= \forall a: A \exists b: B \cdot f a = g b
\end{aligned}$$

Distributive — if  $D \neq \text{null}$

$$\begin{aligned}
&\text{and } v \text{ does not appear in } a \\
a \wedge \forall v: D \cdot b &= \forall v: D \cdot a \wedge b \\
a \wedge \exists v: D \cdot b &= \exists v: D \cdot a \wedge b \\
a \vee \forall v: D \cdot b &= \forall v: D \cdot a \vee b \\
a \vee \exists v: D \cdot b &= \exists v: D \cdot a \vee b \\
a \Rightarrow \forall v: D \cdot b &= \forall v: D \cdot a \Rightarrow b \\
a \Rightarrow \exists v: D \cdot b &= \exists v: D \cdot a \Rightarrow b
\end{aligned}$$

Idempotent — if  $D \neq \text{null}$

$$\begin{aligned}
&\text{and } v \text{ does not appear in } b \\
\forall v: D \cdot b &= b \\
\exists v: D \cdot b &= b
\end{aligned}$$

Absorption — if  $x: D$

$$\begin{aligned}\langle v: D \cdot b \rangle x \wedge \exists v: D \cdot b &= \langle v: D \cdot b \rangle x \\ \langle v: D \cdot b \rangle x \vee \forall v: D \cdot b &= \langle v: D \cdot b \rangle x \\ \langle v: D \cdot b \rangle x \wedge \forall v: D \cdot b &= \forall v: D \cdot b \\ \langle v: D \cdot b \rangle x \vee \exists v: D \cdot b &= \exists v: D \cdot b\end{aligned}$$

Specialization — if element  $x: \square p$

$$\forall p \Rightarrow p x$$

One-Point — if  $x: D$

$$\begin{aligned}\text{and } v \text{ does not appear in } x \\ \forall v: D \cdot v=x \Rightarrow b &= \langle v: D \cdot b \rangle x \\ \exists v: D \cdot v=x \wedge b &= \langle v: D \cdot b \rangle x\end{aligned}$$

Duality

$$\begin{aligned}\neg \forall v \cdot b &= \exists v \cdot \neg b \\ \neg \exists v \cdot b &= \forall v \cdot \neg b \\ \neg \uparrow v \cdot n &= \downarrow v \cdot \neg n \\ \neg \downarrow v \cdot n &= \uparrow v \cdot \neg n\end{aligned}$$

Solution

$$\begin{aligned}\S v: D \cdot \top &= D \\ (\S v: D \cdot b): D & \\ \S v: D \cdot \perp &= \text{null} \\ (\S v \cdot b): (\S v \cdot c) &= \forall v \cdot b \Rightarrow c \\ (\S v \cdot b), (\S v \cdot c) &= \S v \cdot b \vee c \\ (\S v \cdot b) \cdot (\S v \cdot c) &= \S v \cdot b \wedge c \\ x: \S p &= x: \square p \wedge p x \\ \forall f &= (\S f) = (\square f) \\ \exists f &= (\S f) \neq \text{null}\end{aligned}$$

Bounding — if  $D \neq \text{null}$

$$\begin{aligned}\text{and } v \text{ does not appear in } n \\ n > (\uparrow v: D \cdot m) &\Rightarrow (\forall v: D \cdot n > m) \\ n < (\downarrow v: D \cdot m) &\Rightarrow (\forall v: D \cdot n < m) \\ n \geq (\uparrow v: D \cdot m) &= (\forall v: D \cdot n \geq m) \\ n \leq (\downarrow v: D \cdot m) &= (\forall v: D \cdot n \leq m) \\ n \geq (\downarrow v: D \cdot m) &\Leftarrow (\exists v: D \cdot n \geq m) \\ n \leq (\uparrow v: D \cdot m) &\Leftarrow (\exists v: D \cdot n \leq m) \\ n > (\downarrow v: D \cdot m) &= (\exists v: D \cdot n > m) \\ n < (\uparrow v: D \cdot m) &= (\exists v: D \cdot n < m)\end{aligned}$$

Distributive — if  $D \neq \text{null}$  and  $v$  does not appear in  $n$

$$\begin{aligned}n \uparrow (\uparrow v: D \cdot m) &= (\uparrow v: D \cdot n \uparrow m) \\ n \uparrow (\downarrow v: D \cdot m) &= (\downarrow v: D \cdot n \uparrow m) \\ n + (\uparrow v: D \cdot m) &= (\uparrow v: D \cdot n + m) \\ n - (\uparrow v: D \cdot m) &= (\downarrow v: D \cdot n - m) \\ (\uparrow v: D \cdot m) - n &= (\uparrow v: D \cdot m - n) \\ n \geq 0 \Rightarrow n \times (\uparrow v: D \cdot m) &= (\uparrow v: D \cdot n \times m) \\ n \leq 0 \Rightarrow n \times (\uparrow v: D \cdot m) &= (\downarrow v: D \cdot n \times m) \\ n \times (\Sigma v: D \cdot m) &= (\Sigma v: D \cdot n \times m)\end{aligned}$$

Antidistributive — if  $D \neq \text{null}$

$$\begin{aligned}\text{and } v \text{ does not appear in } a \\ a \Leftarrow \exists v: D \cdot b &= \forall v: D \cdot a \Leftarrow b \\ a \Leftarrow \forall v: D \cdot b &= \exists v: D \cdot a \Leftarrow b\end{aligned}$$

Generalization — if element  $x: \square p$

$$p x \Rightarrow \exists p$$

Splitting — for any fixed domain

$$\begin{aligned}\forall v \cdot a \wedge b &= (\forall v \cdot a) \wedge (\forall v \cdot b) \\ \exists v \cdot a \wedge b &\Rightarrow (\exists v \cdot a) \wedge (\exists v \cdot b) \\ \forall v \cdot a \vee b &\Leftarrow (\forall v \cdot a) \vee (\forall v \cdot b) \\ \exists v \cdot a \vee b &= (\exists v \cdot a) \vee (\exists v \cdot b) \\ \forall v \cdot a \Rightarrow b &\Rightarrow (\forall v \cdot a) \Rightarrow (\forall v \cdot b) \\ \forall v \cdot a \Rightarrow b &\Rightarrow (\exists v \cdot a) \Rightarrow (\exists v \cdot b) \\ \forall v \cdot a = b &\Rightarrow (\forall v \cdot a) = (\forall v \cdot b) \\ \forall v \cdot a = b &\Rightarrow (\exists v \cdot a) = (\exists v \cdot b)\end{aligned}$$

Commutative

$$\begin{aligned}\forall v \cdot \forall w \cdot b &= \forall w \cdot \forall v \cdot b \\ \exists v \cdot \exists w \cdot b &= \exists w \cdot \exists v \cdot b\end{aligned}$$

Semicommutative

$$\begin{aligned}\exists v \cdot \forall w \cdot b &\Rightarrow \forall w \cdot \exists v \cdot b \\ \forall x \cdot \exists y \cdot p x y &= \exists f \cdot \forall x \cdot p x (f x)\end{aligned}$$

Domain Change

$$\begin{aligned}A: B \Rightarrow (\forall v: A \cdot b) &\Leftarrow (\forall v: B \cdot b) \\ A: B \Rightarrow (\exists v: A \cdot b) &\Rightarrow (\exists v: B \cdot b) \\ \forall v: A \cdot v: B \Rightarrow p &= \forall v: A \cdot B \cdot p \\ \exists v: A \cdot v: B \wedge p &= \exists v: A \cdot B \cdot p\end{aligned}$$

Extreme

$$\begin{aligned}(\downarrow n: \text{int} \cdot n) &= (\downarrow n: \text{real} \cdot n) = -\infty \\ (\uparrow n: \text{int} \cdot n) &= (\uparrow n: \text{real} \cdot n) = \infty\end{aligned}$$

Connection

$$\begin{aligned}n \leq m &= \forall k \cdot k \leq n \Rightarrow k \leq m \\ n \leq m &= \forall k \cdot k < n \Rightarrow k < m \\ n \leq m &= \forall k \cdot m \leq k \Rightarrow n \leq k \\ n \leq m &= \forall k \cdot m < k \Rightarrow n < k\end{aligned}$$

$$\begin{aligned}n \downarrow (\downarrow v: D \cdot m) &= (\downarrow v: D \cdot n \downarrow m) \\ n \downarrow (\uparrow v: D \cdot m) &= (\uparrow v: D \cdot n \downarrow m) \\ n + (\downarrow v: D \cdot m) &= (\downarrow v: D \cdot n + m) \\ n - (\downarrow v: D \cdot m) &= (\uparrow v: D \cdot n - m) \\ (\downarrow v: D \cdot m) - n &= (\downarrow v: D \cdot m - n) \\ n \geq 0 \Rightarrow n \times (\downarrow v: D \cdot m) &= (\downarrow v: D \cdot n \times m) \\ n \leq 0 \Rightarrow n \times (\downarrow v: D \cdot m) &= (\uparrow v: D \cdot n \times m) \\ (\Pi v: D \cdot m)^n &= (\Pi v: D \cdot m^n)\end{aligned}$$

### 11.3.9 Limits

Let all domains be  $nat$ .

$$\begin{aligned} (\uparrow m \cdot \downarrow n \cdot f(m+n)) &\leq \Downarrow f \leq (\downarrow m \cdot \uparrow n \cdot f(m+n)) \\ \exists m \cdot \forall n \cdot p(m+n) &\implies \Downarrow p \implies \forall m \cdot \exists n \cdot p(m+n) \\ \Downarrow n \cdot n &= \infty \end{aligned}$$

—End of Limits

### 11.3.10 Specifications and Programs

For specifications  $P$ ,  $Q$ ,  $R$ , and  $S$ , and binary  $b$ ,

$$\begin{aligned} ok &= x'=x \wedge y'=y \wedge \dots \\ x:=e &= x'=e \wedge y'=y \wedge \dots \\ P \cdot Q &= \exists x'', y'', \dots \cdot \langle x', y', \dots \rangle P \langle x'', y'', \dots \rangle \wedge \langle x, y, \dots \rangle Q \langle x'', y'', \dots \rangle \\ P \parallel Q &= \exists tP, tQ \cdot \langle t \cdot P \rangle tP \wedge \langle t \cdot Q \rangle tQ \wedge t' = tP \uparrow tQ \\ \text{if } b \text{ then } P \text{ else } Q \text{ fi} &= b \wedge P \vee \neg b \wedge Q = (b \implies P) \wedge (\neg b \implies Q) \\ \text{new } x: T \cdot P &= \exists x, x': T \cdot P \\ \text{frame } x \cdot P &= P \wedge y'=y \wedge \dots \\ \text{while } b \text{ do } P \text{ od} &= t' \geq t \wedge \text{if } b \text{ then } P. t:=t+1. \text{ while } b \text{ do } P \text{ od else } ok \text{ fi} \\ \forall \sigma, \sigma'. \text{if } b \text{ then } P. W \text{ else } ok \text{ fi} &\Leftarrow W \implies \forall \sigma, \sigma'. \text{while } b \text{ do } P \text{ od} \Leftarrow W \end{aligned}$$

To prove  $F m \Leftarrow \text{for } i:=m;..n \text{ do } P \text{ od}$

prove  $F i \Leftarrow i: m;..n \wedge (P. F(i+1))$

and  $F n \Leftarrow ok$

$$\begin{aligned} A m \implies A' n &\Leftarrow \text{for } i:=m;..n \text{ do } i: m;..n \wedge A i \implies A'(i+1) \text{ od} \\ \text{wait until } w &= t:=t \uparrow w \\ \text{assert } b &= \text{if } b \text{ then } ok \text{ else screen! "error". wait until } \infty \text{ fi} \\ \text{ensure } b &= b \wedge ok \end{aligned}$$

$P. (P \text{ value } e)=e$  but do not double-prime or substitute in  $(P \text{ value } e)$

Data transformer  $D$  satisfies  $\forall new \cdot \exists old \cdot D$  and transforms specification  $S$  to

$$\begin{aligned} \forall old \cdot D &\implies \exists old' \cdot D' \wedge S \\ c? &= t:=t \uparrow (\mathcal{J}c_{\mathcal{r}} + (\text{transit time})). \mathcal{r}:=\mathcal{r} + 1 \\ c &= \mathcal{M}c_{\mathcal{r}-1} \\ c!e &= \mathcal{M}c_{\mathcal{w}c}=e \wedge \mathcal{J}c_{\mathcal{w}c}=t \wedge (\mathcal{w}c:=\mathcal{w}c + 1) \\ \sqrt{c} &= \mathcal{J}c_{\mathcal{r}} + (\text{transit time}) \leq t \\ \text{new } x: time \rightarrow T \cdot S &= \exists x: time \rightarrow T \cdot S \\ \text{new } c? T \cdot S &= \exists \mathcal{M}c: \infty * T \cdot \exists \mathcal{J}c: \infty * xnat \cdot \exists \mathcal{r}c, \mathcal{r}'c, \mathcal{w}c, \mathcal{w}'c: xnat \cdot \mathcal{r}c=\mathcal{w}c=0 \wedge S \\ P.ok &= P = ok.P && \text{identity} \\ P.(Q.R) &= (P.Q).R && \text{associativity} \\ P \vee Q.R \vee S &= (P.R) \vee (P.S) \vee (Q.R) \vee (Q.S) && \text{distributivity} \\ \text{if } b \text{ then } P \text{ else } Q \text{ fi}.R &= \text{if } b \text{ then } P.R \text{ else } Q.R \text{ fi} && \text{distributivity (unprimed } b) \\ P.\text{if } b \text{ then } Q \text{ else } R \text{ fi} &= \text{if } P.b \text{ then } P.Q \text{ else } P.R \text{ fi} && \text{distributivity (unprimed } b) \\ P \parallel Q &= Q \parallel P && \text{symmetry} \\ P \parallel (Q \parallel R) &= (P \parallel Q) \parallel R && \text{associativity} \\ P \parallel Q \vee R &= (P \parallel Q) \vee (P \parallel R) && \text{distributivity} \\ P \parallel \text{if } b \text{ then } Q \text{ else } R \text{ fi} &= \text{if } b \text{ then } P \parallel Q \text{ else } P \parallel R \text{ fi} && \text{distributivity} \\ \text{if } b \text{ then } P \parallel Q \text{ else } R \parallel S \text{ fi} &= \text{if } b \text{ then } P \text{ else } R \text{ fi} \parallel \text{if } b \text{ then } Q \text{ else } S \text{ fi} && \text{distributivity} \\ x:=\text{if } b \text{ then } e \text{ else } f \text{ fi} &= \text{if } b \text{ then } x:=e \text{ else } x:=f \text{ fi} && \text{functional-imperative} \end{aligned}$$

—End of Specifications and Programs

### 11.3.11 Substitution

Let  $x$  and  $y$  be different boundary state variables, let  $e$  and  $f$  be expressions of the prestate, and let  $S$  be a specification.

$$x := e. S = (\text{for } x \text{ substitute } e \text{ in } S)$$

$$(x := e \parallel y := f). S = (\text{for } x \text{ substitute } e \text{ and concurrently for } y \text{ substitute } f \text{ in } S)$$

---

End of Substitution

### 11.3.12 Assertions

Let  $P$  and  $Q$  be specifications. Let  $A$  be an assertion and let  $A'$  be the same as  $A$  but with primes on all the variables.

$$A \wedge (P.Q) = A \wedge P.Q$$

$$A \Rightarrow (P.Q) \Leftarrow A \Rightarrow P.Q$$

$$(P.Q) \wedge A' = P.Q \wedge A'$$

$$(P.Q) \Leftarrow A' \Leftarrow P.Q \Leftarrow A'$$

$$P.A \wedge Q = P \wedge A'.Q$$

$$P.Q \Leftarrow P \wedge A'.A \Rightarrow Q$$

$A$  is a sufficient precondition for  $P$  to be refined by  $S$   
if and only if  $A \Rightarrow P$  is refined by  $S$ .

$A'$  is a sufficient postcondition for  $P$  to be refined by  $S$   
if and only if  $A' \Rightarrow P$  is refined by  $S$ .

---

End of Assertions

### 11.3.13 Refinement

Refinement by Steps (Stepwise Refinement) (monotonicity, transitivity)

If  $A \Leftarrow \mathbf{if } b \mathbf{ then } C \mathbf{ else } D \mathbf{ fi}$  and  $C \Leftarrow E$  and  $D \Leftarrow F$  are theorems,  
then  $A \Leftarrow \mathbf{if } b \mathbf{ then } E \mathbf{ else } F \mathbf{ fi}$  is a theorem.

If  $A \Leftarrow B.C$  and  $B \Leftarrow D$  and  $C \Leftarrow E$  are theorems, then  $A \Leftarrow D.E$  is a theorem.

If  $A \Leftarrow B \parallel C$  and  $B \Leftarrow D$  and  $C \Leftarrow E$  are theorems, then  $A \Leftarrow D \parallel E$  is a theorem.

If  $A \Leftarrow B$  and  $B \Leftarrow C$  are theorems, then  $A \Leftarrow C$  is a theorem.

Refinement by Parts (monotonicity, conflation)

If  $A \Leftarrow \mathbf{if } b \mathbf{ then } C \mathbf{ else } D \mathbf{ fi}$  and  $E \Leftarrow \mathbf{if } b \mathbf{ then } F \mathbf{ else } G \mathbf{ fi}$  are theorems,  
then  $A \wedge E \Leftarrow \mathbf{if } b \mathbf{ then } C \wedge F \mathbf{ else } D \wedge G \mathbf{ fi}$  is a theorem.

If  $A \Leftarrow B.C$  and  $D \Leftarrow E.F$  are theorems, then  $A \wedge D \Leftarrow B \wedge E. C \wedge F$  is a theorem.

If  $A \Leftarrow B \parallel C$  and  $D \Leftarrow E \parallel F$  are theorems, then  $A \wedge D \Leftarrow B \wedge E \parallel C \wedge F$  is a theorem.

If  $A \Leftarrow B$  and  $C \Leftarrow D$  are theorems, then  $A \wedge C \Leftarrow B \wedge D$  is a theorem.

Refinement by Cases

$P \Leftarrow \mathbf{if } b \mathbf{ then } Q \mathbf{ else } R \mathbf{ fi}$  is a theorem if and only if  
 $P \Leftarrow b \wedge Q$  and  $P \Leftarrow \neg b \wedge R$  are theorems.

---

End of Refinement

End of Laws

## 11.4 Names

*abs*:  $xreal \rightarrow (\S r: xreal \cdot r \geq 0)$

*bin* (the binary values)

*ceil*:  $real \rightarrow int$

*char* (the characters)

*div*:  $real \rightarrow (\S r: real \cdot r > 0) \rightarrow int$

*divides*:  $(nat+1) \rightarrow int \rightarrow bin$

*entro*:  $prob \rightarrow (\S r: xreal \cdot r \geq 0)$

*even*:  $int \rightarrow bin$

*floor*:  $real \rightarrow int$

*info*:  $prob \rightarrow (\S r: xreal \cdot r \geq 0)$

*int* (the integers)

*log*:  $(\S r: xreal \cdot r \geq 0) \rightarrow xreal$

*mod*:  $real \rightarrow (\S r: real \cdot r > 0) \rightarrow real$

*nat* (the naturals)

*nil* (the empty string)

*null* (the empty bunch)

*odd*:  $int \rightarrow bin$

*ok* (the empty program)

*prob* (probability)

*rand* (random number)

*rat* (the rationals)

*real* (the reals)

*suc*:  $nat \rightarrow (nat+1)$

*time* (time)

*xint* (the extended integers)

*xnat* (the extended naturals)

*xrat* (the extended rationals)

*xreal* (the extended reals)

*abs*  $r = \mathbf{if} \ r \geq 0 \ \mathbf{then} \ r \ \mathbf{else} \ -r \ \mathbf{fi}$

*bin*  $= \top, \perp$

$r \leq \mathit{ceil} \ r < r+1$

*char*  $= \dots, \text{"a"}, \text{"A"}, \dots$

$\mathit{div} \ x \ y = \mathit{floor} \ (x/y)$

*divides*  $n \ i = i/n: int$

*entro*  $p = p \times \mathit{info} \ p + (1-p) \times \mathit{info} \ (1-p)$

*even*  $i = i/2: int$

*even*  $= \mathit{divides} \ 2$

*floor*  $r \leq r < \mathit{floor} \ r + 1$

*info*  $p = -\log p$

*int*  $= nat, -nat$

*log*  $(2^x) = x$

*log*  $(x \times y) = \log x + \log y$

$0 \leq \mathit{mod} \ a \ d < d$

$a = \mathit{div} \ a \ d \times d + \mathit{mod} \ a \ d$

$0, nat+1: nat$

$0, B+1: B \Rightarrow nat: B$

$\Leftrightarrow nil = 0$

*nil*;  $S = S = S; nil$

*nil*  $\leq S$

$\emptyset null = 0$

*null*,  $A = A = A, null$

*null*:  $A$

*odd*  $i = \neg i/2: int$

*odd*  $= \neg \mathit{even}$

*ok*  $= \sigma' = \sigma$

*ok.P*  $= P = P.ok$

*prob*  $= \S r: real \cdot 0 \leq r \leq 1$

*rand*  $n: 0, ..n$

*rat*  $= int/(nat+1)$

*real*  $= xreal_{\neg(\infty, -\infty)}$

*suc*  $n = n+1$

either *time*  $= xnat$  or *time*  $= \S r: xreal \cdot r \geq 0$

*xint*  $= -\infty, int, \infty$

*xnat*  $= nat, \infty$

*xrat*  $= -\infty, rat, \infty$

$x: xreal = \exists f: nat \rightarrow rat \cdot x = \Downarrow f$

## 11.5 Symbols

symbol	page	pronunciation	symbol	page	pronunciation
$\top$	<a href="#">3</a>	top, true	$\surd$	<a href="#">137</a>	input check
$\perp$	<a href="#">3</a>	bottom, false	$()$	<a href="#">4</a>	precedence brackets
$\neg$	<a href="#">3</a>	not	$\{\}$	<a href="#">17</a>	set brackets
$\wedge$	<a href="#">3</a>	and	$[\ ]$	<a href="#">20</a>	list brackets
$\vee$	<a href="#">3</a>	or	$\langle \rangle$	<a href="#">23</a>	function brackets
$\Rightarrow$	<a href="#">3</a>	implies	$\zeta$	<a href="#">17</a>	power
$\implies$	<a href="#">4</a>	implies	$\#$	<a href="#">14</a>	bunch size, cardinality
$\Leftarrow$	<a href="#">3</a>	follows from, is implied by	$\$$	<a href="#">17</a>	set size, cardinality
$\Leftarrow\Leftarrow$	<a href="#">4</a>	follows from, is implied by	$\leftrightarrow$	<a href="#">17</a>	string size (length)
$=$	<a href="#">3</a>	equals, if and only if	$\#$	<a href="#">20,23</a>	list size (length), function size
$\equiv$	<a href="#">4</a>	equals, if and only if	$ $	<a href="#">20,24</a>	otherwise, selective union
$\neq$	<a href="#">3</a>	differs from, is unequal to	$\parallel$	<a href="#">121</a>	concurrent (parallel) composition
$<$	<a href="#">13</a>	less than	$\sim$	<a href="#">17,20</a>	contents of a set or list
$>$	<a href="#">13</a>	greater than	$*$	<a href="#">18</a>	repetition of a string
$\leq$	<a href="#">13</a>	less than or equal to	$\square$	<a href="#">20,23</a>	domain of a list or function
$\geq$	<a href="#">13</a>	greater than or equal to	$\rightarrow$	<a href="#">25</a>	function arrow
$+$	<a href="#">12</a>	plus	$\in$	<a href="#">17</a>	element of a set
$-$	<a href="#">12</a>	minus	$\subseteq$	<a href="#">17</a>	subset
$\times$	<a href="#">12</a>	times, multiplication	$\cup$	<a href="#">17</a>	set union
$/$	<a href="#">12</a>	divided by	$\cap$	<a href="#">17</a>	set intersection
$\uparrow$	<a href="#">12</a>	maximum	$@$	<a href="#">22</a>	index with a pointer
$\downarrow$	<a href="#">12</a>	minimum	$\forall$	<a href="#">26</a>	for all, universal quantifier
$,$	<a href="#">14</a>	bunch union	$\exists$	<a href="#">26</a>	there exists, existential quantifier
$,..$	<a href="#">16</a>	union from (including) to (excluding)	$\Sigma$	<a href="#">26</a>	sum of, summation quantifier
$'$	<a href="#">14</a>	bunch intersection	$\Pi$	<a href="#">26</a>	product of, product quantifier
$;$	<a href="#">17</a>	string join	$\Uparrow$	<a href="#">26</a>	maximum (least upper bound) quantifier
$::$	<a href="#">20</a>	list join	$\Downarrow$	<a href="#">26</a>	minimum (greatest lower bound) quantifier
$;..$	<a href="#">19</a>	join from (including) to (excluding)	$\Updownarrow$	<a href="#">33</a>	limit quantifier
$:$	<a href="#">14</a>	is in, are in, bunch inclusion	$\S$	<a href="#">28</a>	those, solution quantifier
$::$	<a href="#">14</a>	includes	$'$	<a href="#">34</a>	$x'$ is final value of state variable $x$
$:=$	<a href="#">36</a>	assignment	$“ ”$	<a href="#">13,19</a>	“hi” is a text or string of characters
$\otimes$	<a href="#">78</a>	label, target of <b>go to</b>	$a^b$	<a href="#">12</a>	exponentiation
$.$	<a href="#">36</a>	sequential composition	$a_b$	<a href="#">18</a>	string indexing
$\cdot$	<a href="#">23</a>	function and quantifier	$a b$	<a href="#">20,24,31</a>	indexing, application, composition
$\bar{\_}$	<a href="#">14</a>	bunch removal	$\triangleleft \triangleright$	<a href="#">18</a>	string modification
$!$	<a href="#">137</a>	output	$\infty$	<a href="#">12</a>	infinity
$?$	<a href="#">137</a>	input			
<b>assert</b>	<a href="#">79</a>		<b>if then else fi</b>	<a href="#">4</a>	
<b>do od</b>	<a href="#">73</a>		<b>new</b>	<a href="#">68</a>	
<b>ensure</b>	<a href="#">80</a>		<b>or</b>	<a href="#">80</a>	
<b>exit when</b>	<a href="#">73</a>		<b>value</b>	<a href="#">81</a>	
<b>for do od</b>	<a href="#">76</a>		<b>wait until</b>	<a href="#">79</a>	
<b>frame</b>	<a href="#">69</a>		<b>while do od</b>	<a href="#">71</a>	
<b>go to</b>	<a href="#">78</a>				

