

Systematic Programming

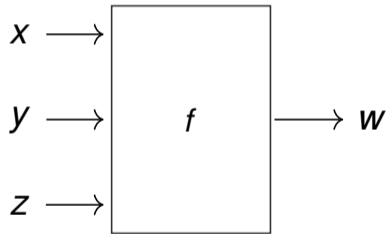
Functions and Expressions

Dr. Maurice Chandoo

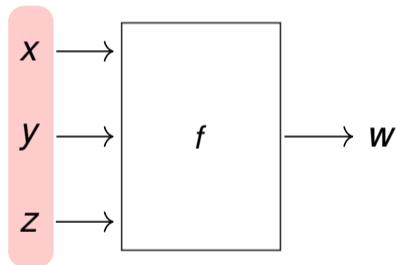
Leibniz Universität Hannover
Institut für Theoretische Informatik

Summer Semester 2020

Functions in Programming

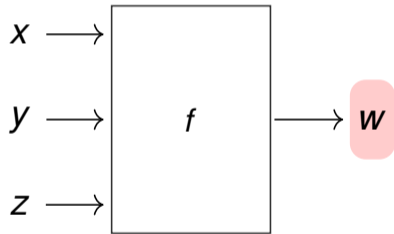


Functions in Programming



input parameters

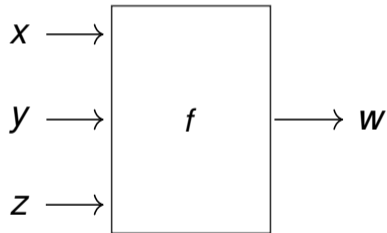
Functions in Programming



input parameters

return value

Functions in Programming

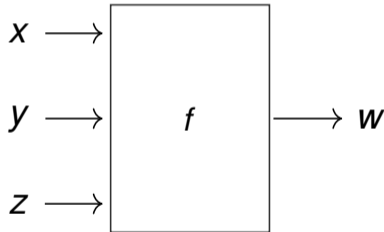


input parameters

return value

Functions in Programming

- ▶ functions are **unambiguous**

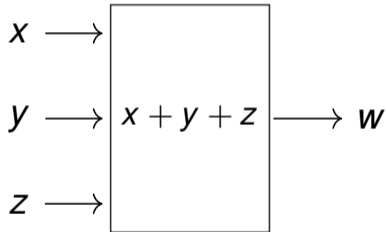


input parameters

return value

Functions in Programming

- ▶ functions are **unambiguous**

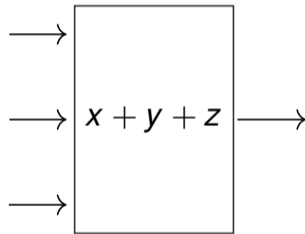


input parameters

return value

Functions in Programming

- ▶ functions are **unambiguous**

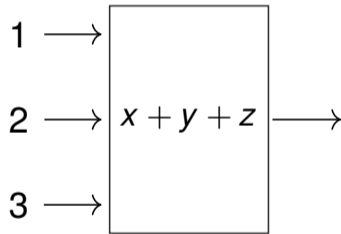


input parameters

return value

Functions in Programming

- ▶ functions are **unambiguous**

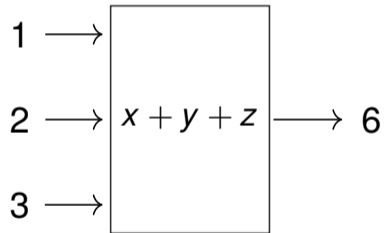


input parameters

return value

Functions in Programming

- ▶ functions are **unambiguous**

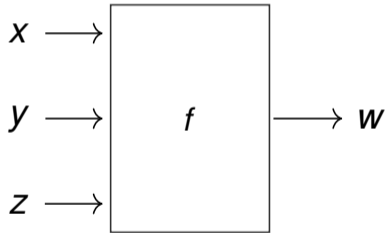


input parameters

return value

Functions in Programming

- ▶ functions are **unambiguous**

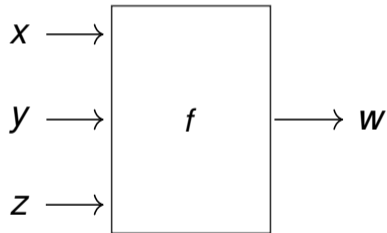


input parameters

return value

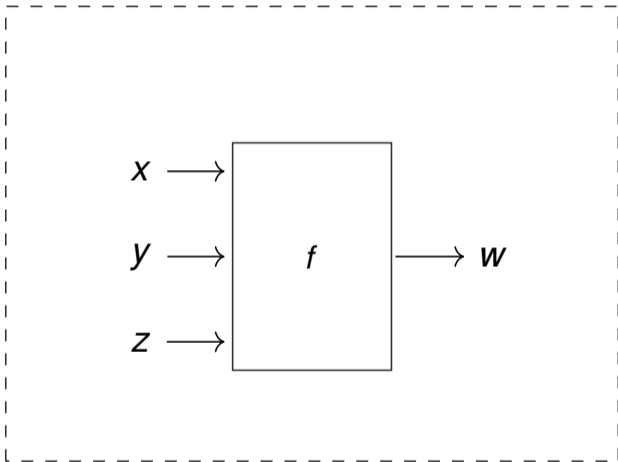
Functions in Programming

- ▶ functions are **unambiguous**



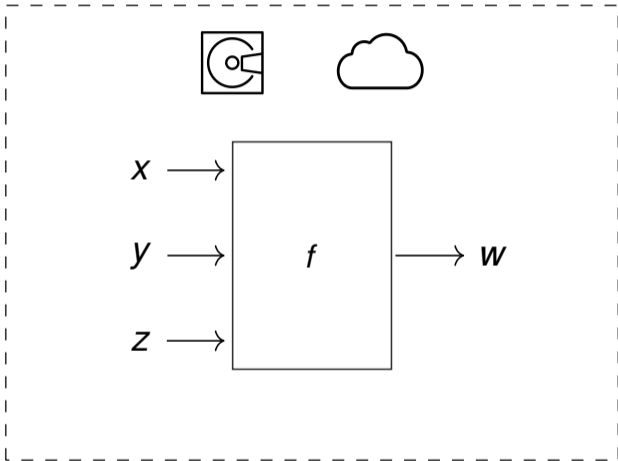
Functions in Programming

- ▶ functions are **unambiguous**
- ▶ execution environment



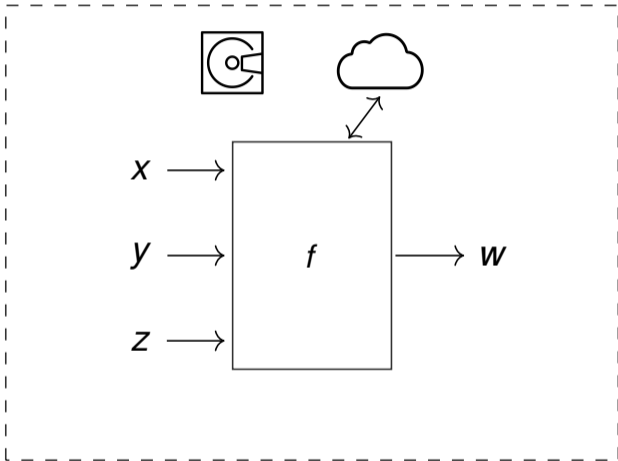
Functions in Programming

- ▶ functions are **unambiguous**
- ▶ execution environment
HDD, network connections, ...



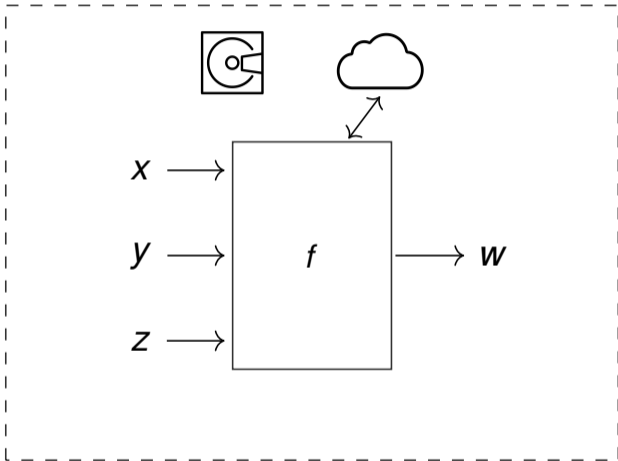
Functions in Programming

- ▶ functions are **unambiguous**
- ▶ execution environment
HDD, network connections, ...



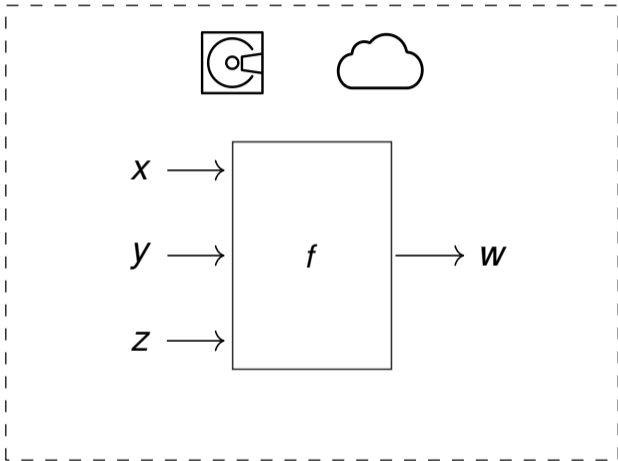
Functions in Programming

- ▶ functions are **unambiguous**
- ▶ execution environment
HDD, network connections, ...
- ▶ interaction with environment can
violate unambiguity



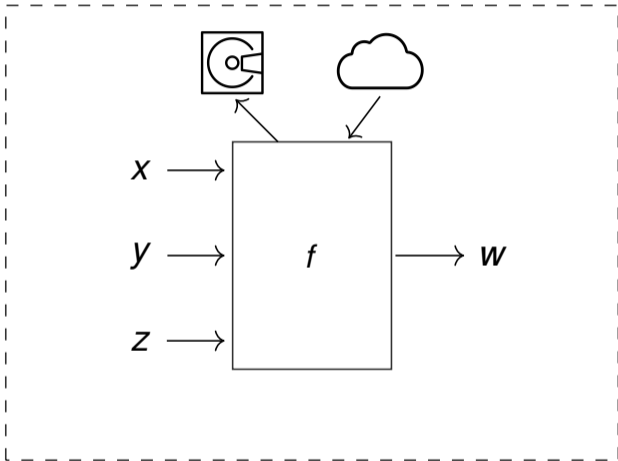
Functions in Programming

- ▶ functions are **unambiguous**
- ▶ execution environment
HDD, network connections, ...
- ▶ interaction with environment can
violate unambiguity



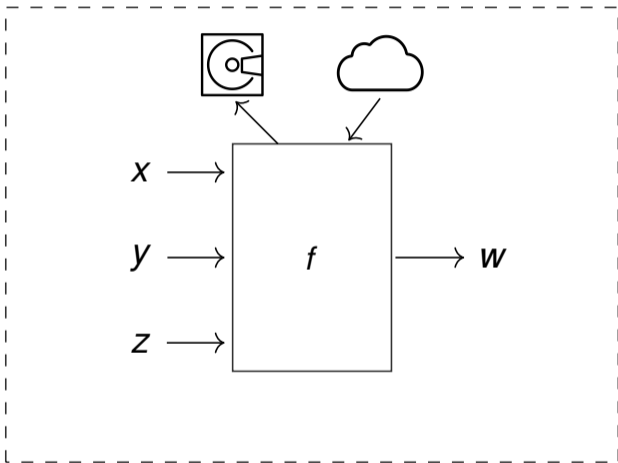
Functions in Programming

- ▶ functions are **unambiguous**
- ▶ execution environment
HDD, network connections, ...
- ▶ interaction with environment can
violate unambiguity



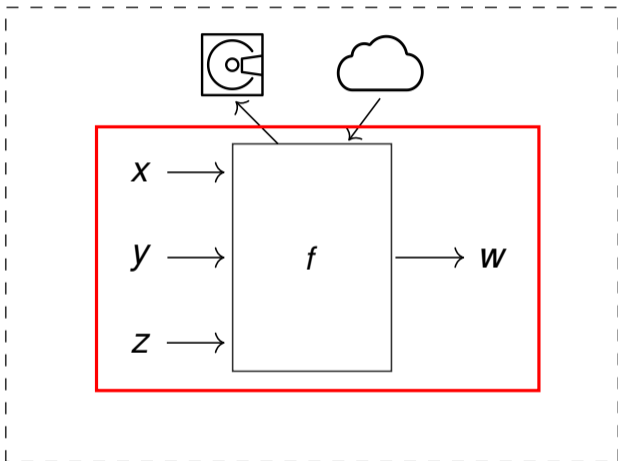
Functions in Programming

- ▶ functions are **unambiguous**
- ▶ execution environment
HDD, network connections, ...
- ▶ interaction with environment can
violate unambiguity
- ▶ **side-effect**
modification of environment



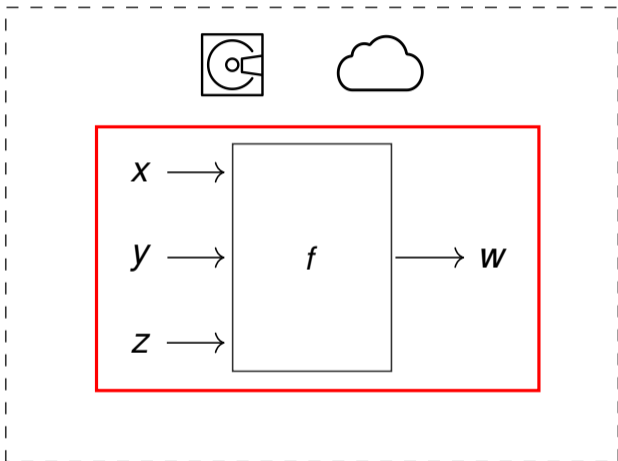
Functions in Programming

- ▶ functions are **unambiguous**
- ▶ execution environment
HDD, network connections, ...
- ▶ interaction with environment can
violate unambiguity
- ▶ **side-effect**
modification of environment



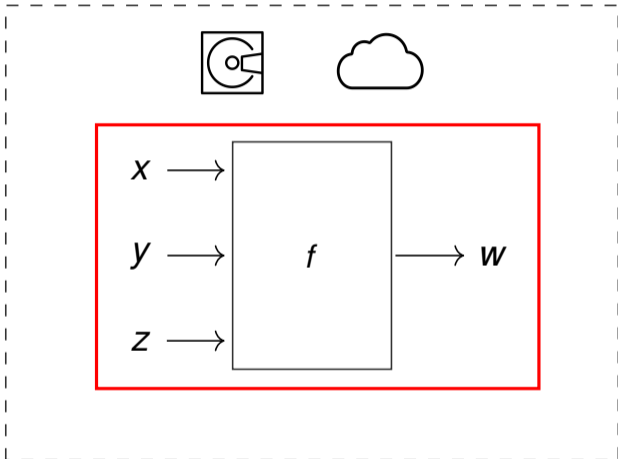
Functions in Programming

- ▶ functions are **unambiguous**
- ▶ execution environment
HDD, network connections, ...
- ▶ interaction with environment can
violate unambiguity
- ▶ **side-effect**
modification of environment



Functions in Programming

- ▶ functions are **unambiguous**
- ▶ execution environment
HDD, network connections, ...
- ▶ interaction with environment can
violate unambiguity
- ▶ **side-effect**
modification of environment
- ▶ **pure function**
no interaction with environment



Describing Functions

- ▶ $f(n) = \text{largest prime number } \leq n$

Describing Functions

- ▶ $f(n) = \text{largest prime number } \leq n$

n	$f(n)$
2	2
3	3
4	3
5	5
6	5

Describing Functions

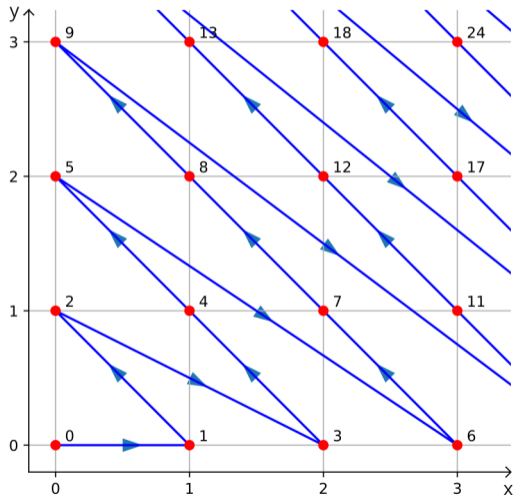
- ▶ $f(n) = \text{largest prime number } \leq n$

Describing Functions

- ▶ $f(n) = \text{largest prime number } \leq n$
- ▶ $c(x, y)$

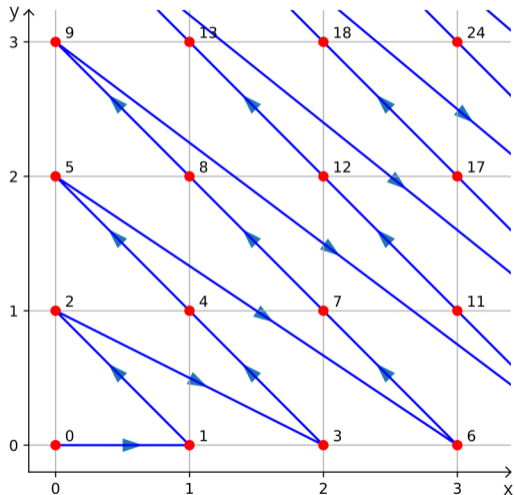
Describing Functions

- ▶ $f(n) = \text{largest prime number } \leq n$
- ▶ $c(x, y)$



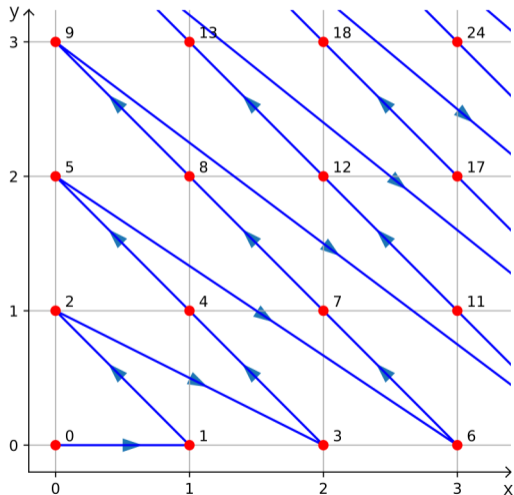
Describing Functions

- ▶ $f(n) =$ largest prime number $\leq n$
- ▶ $c(x, y) =$ no. of steps until (x, y)



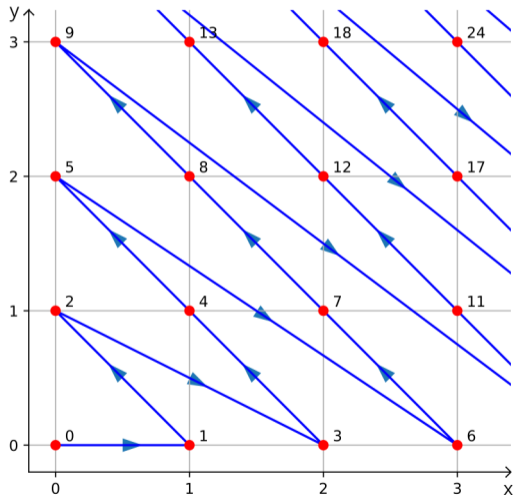
Describing Functions

- ▶ $f(n) = \text{largest prime number } \leq n$
- ▶ $c(x, y) = \text{no. of steps until } (x, y)$
- ▶ $c(x, y) = y + \frac{(x+y)(x+y+1)}{2}$



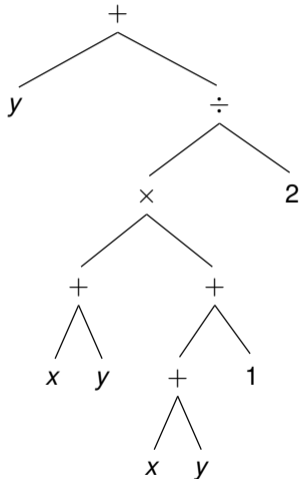
Describing Functions

- ▶ $f(n) = \text{largest prime number } \leq n$
- ▶ $c(x, y) = \text{no. of steps until } (x, y)$
- ▶ $c(x, y) = y + \frac{(x+y)(x+y+1)}{2}$
- ▶ expressions describe functions



Describing Functions

- ▶ $f(n) = \text{largest prime number } \leq n$
- ▶ $c(x, y) = \text{no. of steps until } (x, y)$
- ▶ $c(x, y) = y + \frac{(x+y)(x+y+1)}{2}$
- ▶ expressions describe functions

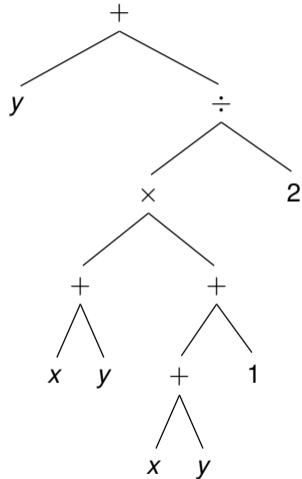


Expressions

- ▶ **expression**
tree labeled with variables & functions

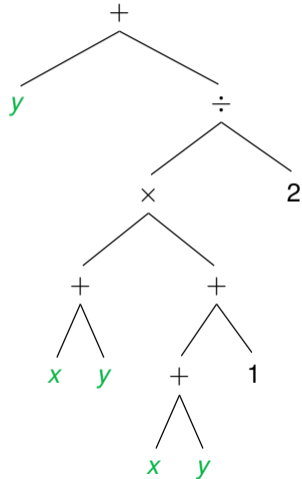
Expressions

- ▶ **expression**
tree labeled with variables & functions



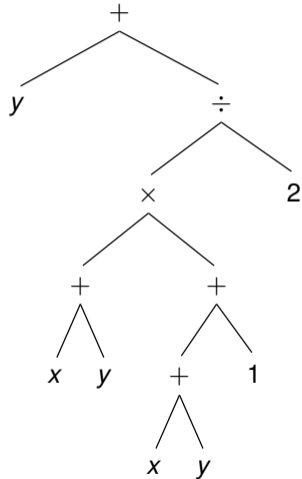
Expressions

- ▶ **expression**
tree labeled with variables & functions



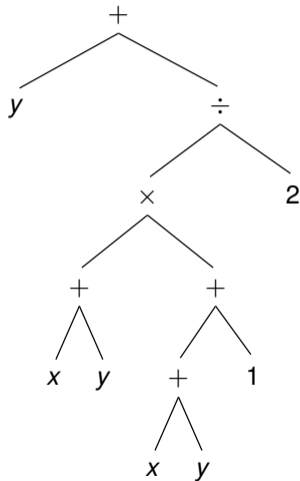
Expressions

- ▶ **expression**
tree labeled with variables & functions



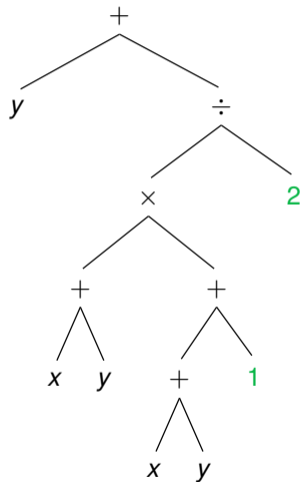
Expressions

- ▶ **expression**
tree labeled with variables & functions
- ▶ **constant**
function without input parameters



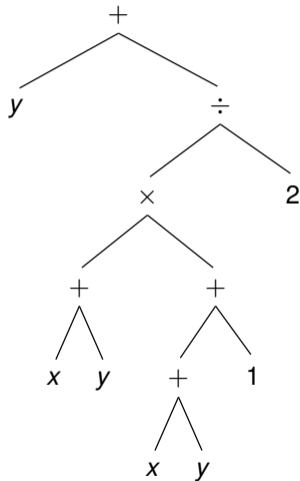
Expressions

- ▶ **expression**
tree labeled with variables & functions
- ▶ **constant**
function without input parameters



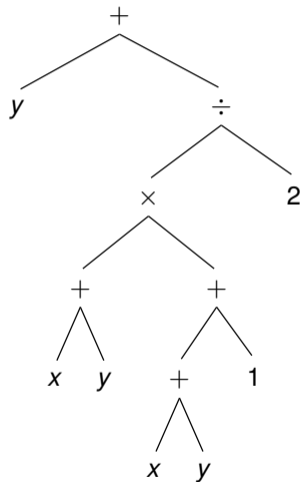
Expressions

- ▶ **expression**
tree labeled with variables & functions
- ▶ **constant**
function without input parameters



Expressions

- ▶ **expression**
tree labeled with variables & functions
- ▶ **constant**
function without input parameters
- ▶ **basis functions**
set of given functions



Expressions

- ▶ **expression**
tree labeled with variables & functions
- ▶ **constant**
function without input parameters
- ▶ **basis functions**
set of given functions

$$f(x) = g(x) + h(x)$$

$$g(x) = x \times h(2)$$

$$h(x) = x + 1$$

Expressions

- ▶ **expression**
tree labeled with variables & functions
- ▶ **constant**
function without input parameters
- ▶ **basis functions**
set of given functions

$$f(x) = g(x) + h(x)$$

$$g(x) = x \times h(2)$$

$$h(x) = x + 1$$

call graph

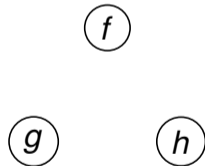
Expressions

- ▶ **expression**
tree labeled with variables & functions
- ▶ **constant**
function without input parameters
- ▶ **basis functions**
set of given functions

$$f(x) = g(x) + h(x)$$

$$g(x) = x \times h(2)$$

$$h(x) = x + 1$$



call graph

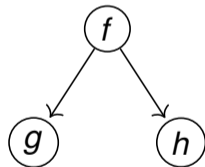
Expressions

- ▶ **expression**
tree labeled with variables & functions
- ▶ **constant**
function without input parameters
- ▶ **basis functions**
set of given functions

$$f(x) = g(x) + h(x)$$

$$g(x) = x \times h(2)$$

$$h(x) = x + 1$$



call graph

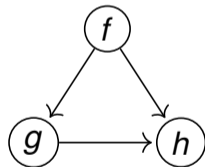
Expressions

- ▶ **expression**
tree labeled with variables & functions
- ▶ **constant**
function without input parameters
- ▶ **basis functions**
set of given functions

$$f(x) = g(x) + h(x)$$

$$g(x) = x \times h(2)$$

$$h(x) = x + 1$$



call graph

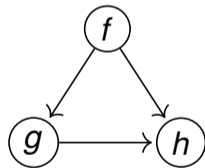
Expressions

- ▶ **expression**
tree labeled with variables & functions
- ▶ **constant**
function without input parameters
- ▶ **basis functions**
set of given functions
- ▶ an expression is **recursive** if its function in the call graph leads to a cycle

$$f(x) = g(x) + h(x)$$

$$g(x) = x \times h(2)$$

$$h(x) = x + 1$$



call graph

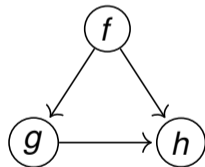
Expressions

- ▶ **expression**
tree labeled with variables & functions
- ▶ **constant**
function without input parameters
- ▶ **basis functions**
set of given functions
- ▶ an expression is **recursive** if its function in the call graph leads to a cycle

$$f(x) = g(x) + h(x)$$

$$g(x) = x \times h(2)$$

$$h(x) = g(x)$$



call graph

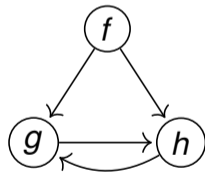
Expressions

- ▶ **expression**
tree labeled with variables & functions
- ▶ **constant**
function without input parameters
- ▶ **basis functions**
set of given functions
- ▶ an expression is **recursive** if its function in the call graph leads to a cycle

$$f(x) = g(x) + h(x)$$

$$g(x) = x \times h(2)$$

$$h(x) = g(x)$$



call graph

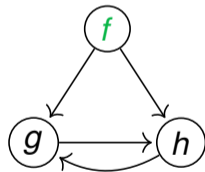
Expressions

- ▶ **expression**
tree labeled with variables & functions
- ▶ **constant**
function without input parameters
- ▶ **basis functions**
set of given functions
- ▶ an expression is **recursive** if its function in the call graph leads to a cycle

$$f(x) = g(x) + h(x)$$

$$g(x) = x \times h(2)$$

$$h(x) = g(x)$$



call graph

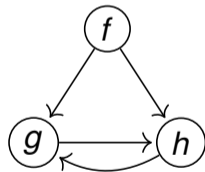
Expressions

- ▶ **expression**
tree labeled with variables & functions
- ▶ **constant**
function without input parameters
- ▶ **basis functions**
set of given functions
- ▶ an expression is **recursive** if its function in the call graph leads to a cycle

$$f(x) = g(x) + h(x)$$

$$g(x) = x \times h(2)$$

$$h(x) = g(x)$$



call graph

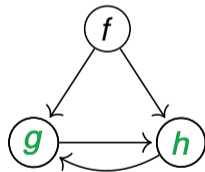
Expressions

- ▶ **expression**
tree labeled with variables & functions
- ▶ **constant**
function without input parameters
- ▶ **basis functions**
set of given functions
- ▶ an expression is **recursive** if its function in the call graph leads to a cycle

$$f(x) = g(x) + h(x)$$

$$g(x) = x \times h(2)$$

$$h(x) = g(x)$$



call graph

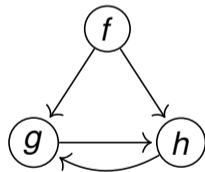
Expressions

- ▶ **expression**
tree labeled with variables & functions
- ▶ **constant**
function without input parameters
- ▶ **basis functions**
set of given functions
- ▶ an expression is **recursive** if its function in the call graph leads to a cycle

$$f(x) = g(x) + h(x)$$

$$g(x) = x \times h(2)$$

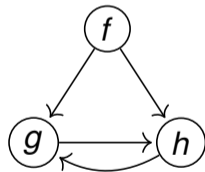
$$h(x) = g(x)$$



call graph

Expressions

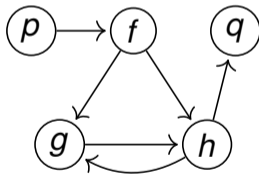
- ▶ **expression**
tree labeled with variables & functions
- ▶ **constant**
function without input parameters
- ▶ **basis functions**
set of given functions
- ▶ an expression is **recursive** if its function in the call graph leads to a cycle



call graph

Expressions

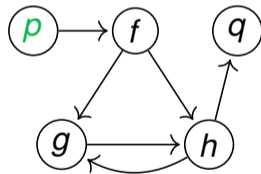
- ▶ **expression**
tree labeled with variables & functions
- ▶ **constant**
function without input parameters
- ▶ **basis functions**
set of given functions
- ▶ an expression is **recursive** if its function in the call graph leads to a cycle



call graph

Expressions

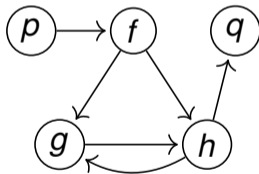
- ▶ **expression**
tree labeled with variables & functions
- ▶ **constant**
function without input parameters
- ▶ **basis functions**
set of given functions
- ▶ an expression is **recursive** if its function in the call graph leads to a cycle



call graph

Expressions

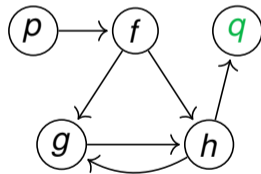
- ▶ **expression**
tree labeled with variables & functions
- ▶ **constant**
function without input parameters
- ▶ **basis functions**
set of given functions
- ▶ an expression is **recursive** if its function in the call graph leads to a cycle



call graph

Expressions

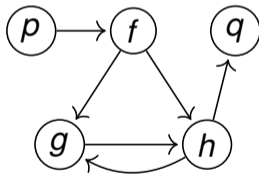
- ▶ **expression**
tree labeled with variables & functions
- ▶ **constant**
function without input parameters
- ▶ **basis functions**
set of given functions
- ▶ an expression is **recursive** if its function in the call graph leads to a cycle



call graph

Expressions

- ▶ **expression**
tree labeled with variables & functions
- ▶ **constant**
function without input parameters
- ▶ **basis functions**
set of given functions
- ▶ an expression is **recursive** if its function in the call graph leads to a cycle



call graph

Recursive Expression

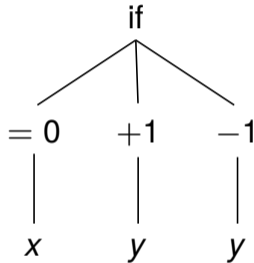
- ▶ find expression for addition

Recursive Expression

- ▶ find expression for addition
- ▶ basis functions: $+1$, -1 , $= 0$, if

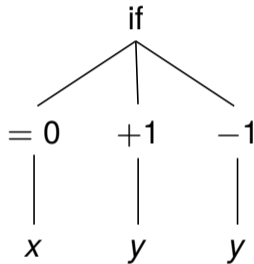
Recursive Expression

- ▶ find expression for addition
- ▶ basis functions: $+1$, -1 , $= 0$, if



Recursive Expression

- ▶ find expression for addition
- ▶ basis functions: $+1$, -1 , $= 0$, if



$$(x, y) \mapsto \begin{cases} y + 1, & \text{if } x = 0 \\ y - 1, & \text{if } x \neq 0 \end{cases}$$

Recursive Expression

- ▶ find expression for addition
- ▶ basis functions: $+1$, -1 , $= 0$, if

Recursive Expression

- ▶ find expression for addition
- ▶ basis functions: $+1$, -1 , $= 0$, if
- ▶ expression for $\text{add}(x, y)$

Recursive Expression

- ▶ find expression for addition
- ▶ basis functions: $+1$, -1 , $= 0$, if
- ▶ expression for $\text{add}(x, y)$

$$\text{add}(x, y) = x + \underbrace{1 + \dots + 1}_{y \text{ times}}$$

Recursive Expression

- ▶ find expression for addition
- ▶ basis functions: $+1$, -1 , $= 0$, if
- ▶ expression for $\text{add}(x, y)$
- ▶ recursive expression necessary

$$\text{add}(x, y) = x + \underbrace{1 + \dots + 1}_{y \text{ times}}$$

Recursive Expression

- ▶ find expression for addition
- ▶ basis functions: $+1$, -1 , $= 0$, if
- ▶ expression for $\text{add}(x, y)$
- ▶ recursive expression necessary

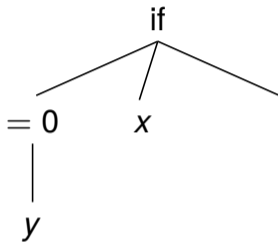
Recursive Expression

- ▶ find expression for addition
- ▶ basis functions: $+1$, -1 , $= 0$, if
- ▶ expression for $\text{add}(x, y)$
- ▶ recursive expression necessary

$$x + 0 = x$$

Recursive Expression

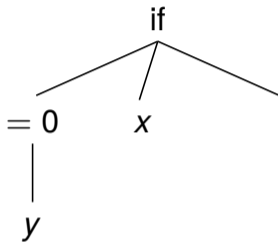
- ▶ find expression for addition
- ▶ basis functions: $+1$, -1 , $= 0$, if
- ▶ expression for $\text{add}(x, y)$
- ▶ recursive expression necessary



$$x + 0 = x$$

Recursive Expression

- ▶ find expression for addition
- ▶ basis functions: $+1$, -1 , $= 0$, `if`
- ▶ expression for `add(x, y)`
- ▶ recursive expression necessary

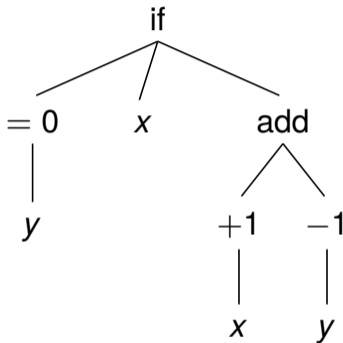


$$x + 0 = x$$

$$x + y = (x + 1) + (y - 1)$$

Recursive Expression

- ▶ find expression for addition
- ▶ basis functions: $+1$, -1 , $= 0$, if
- ▶ expression for $\text{add}(x, y)$
- ▶ recursive expression necessary

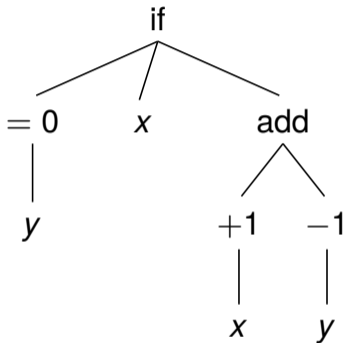


$$x + 0 = x$$

$$x + y = (x + 1) + (y - 1)$$

Recursive Expression

- ▶ find expression for addition
- ▶ basis functions: $+1$, -1 , $= 0$, if
- ▶ expression for $\text{add}(x, y)$
- ▶ recursive expression necessary
- ▶ ex.: generalize to \mathbb{Z}

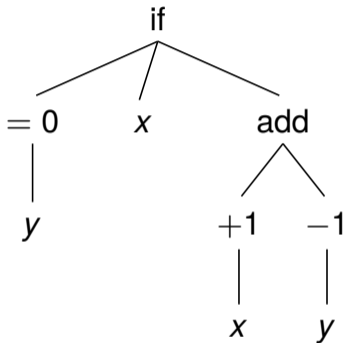


$$x + 0 = x$$

$$x + y = (x + 1) + (y - 1)$$

Recursive Expression

- ▶ find expression for addition
- ▶ basis functions: $+1$, -1 , $= 0$, if
- ▶ expression for $\text{add}(x, y)$
- ▶ recursive expression necessary
- ▶ ex.: generalize to \mathbb{Z}
additional basis function < 0

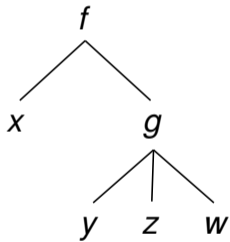


$$x + 0 = x$$

$$x + y = (x + 1) + (y - 1)$$

Representations for Expressions

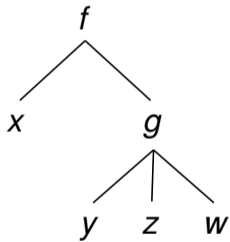
tree



Representations for Expressions

tree

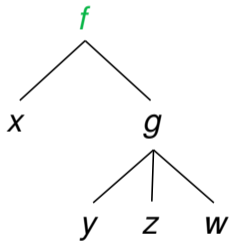
prefix notation



Representations for Expressions

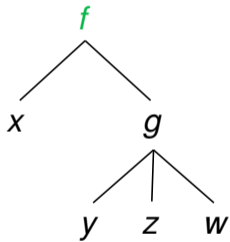
tree

prefix notation



Representations for Expressions

tree

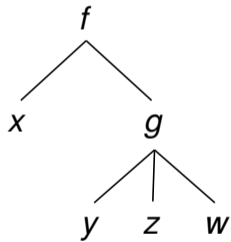


prefix notation

()

Representations for Expressions

tree

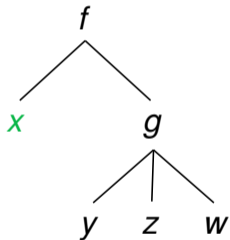


prefix notation

()

Representations for Expressions

tree

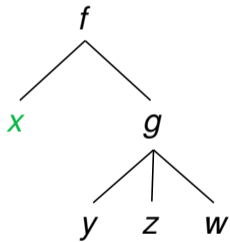


prefix notation

()

Representations for Expressions

tree

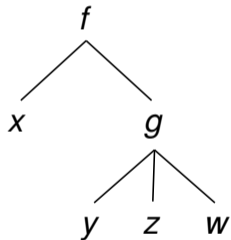


prefix notation

(())

Representations for Expressions

tree

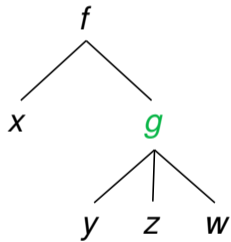


prefix notation

(())

Representations for Expressions

tree

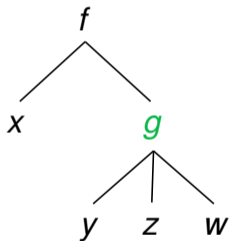


prefix notation

(())

Representations for Expressions

tree

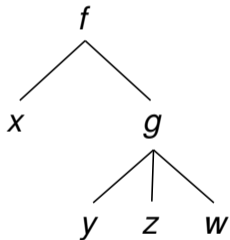


prefix notation

(() ())

Representations for Expressions

tree

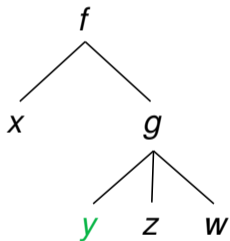


prefix notation

$((()()))$

Representations for Expressions

tree

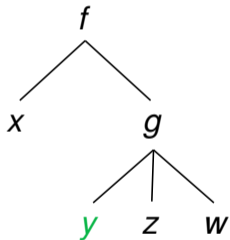


prefix notation

$((() (\quad)))$

Representations for Expressions

tree

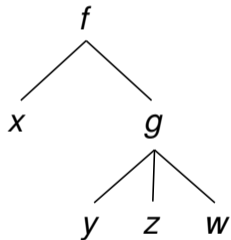


prefix notation

(() (()))

Representations for Expressions

tree

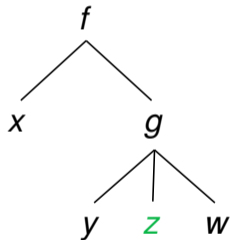


prefix notation

$((()()))$

Representations for Expressions

tree

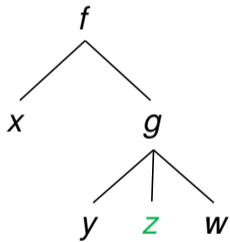


prefix notation

(() (()))

Representations for Expressions

tree

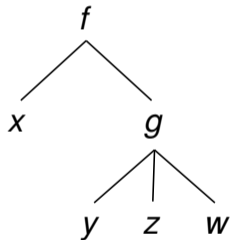


prefix notation

(() (() ()))

Representations for Expressions

tree

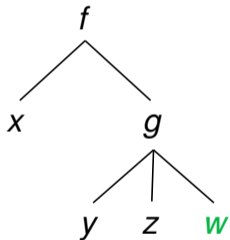


prefix notation

$((()((())())))$

Representations for Expressions

tree

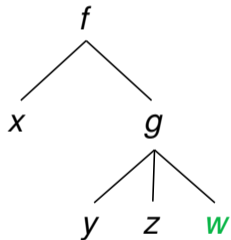


prefix notation

(() (() ()))

Representations for Expressions

tree

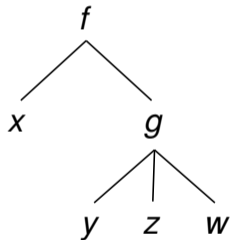


prefix notation

(() (() () ())))

Representations for Expressions

tree

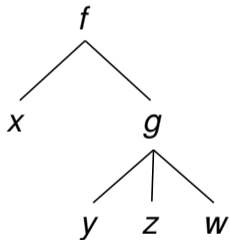


prefix notation

(() (() () ()))

Representations for Expressions

tree

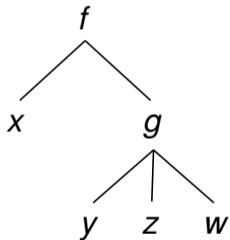


prefix notation

$(f(x)(g(y)(z)(w)))$

Representations for Expressions

tree



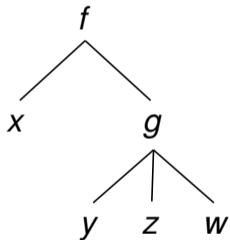
prefix notation

$(f(x)(g(y)(z)(w)))$



Representations for Expressions

tree



prefix notation

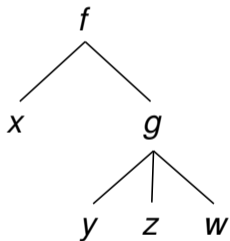
$(f(x)(g(y)(z)(w)))$



$(f(x)(g(y)(z)(w)))$

Representations for Expressions

tree



prefix notation

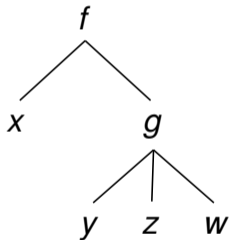
$(f(x)(g(y)(z)(w)))$



$(f\ x\ (g\ y\ z\ w\))$

Representations for Expressions

tree



prefix notation

$(f(x)(g(y)(z)(w)))$

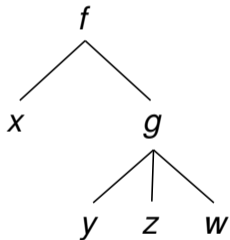


$(f\ x\ (g\ y\ z\ w\))$

math. notation

Representations for Expressions

tree



prefix notation

$(f(x)(g(y)(z)(w)))$



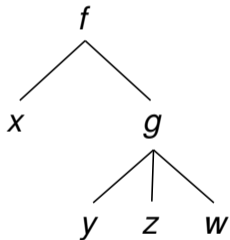
$(f x (g y z w))$

math. notation



Representations for Expressions

tree



prefix notation

$(f(x)(g(y)(z)(w)))$



$(f x (g y z w))$

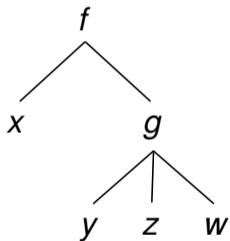
math. notation

$(f x (g y z w))$



Representations for Expressions

tree



prefix notation

$(f(x)(g(y)(z)(w)))$



$(f\ x\ (g\ y\ z\ w\))$

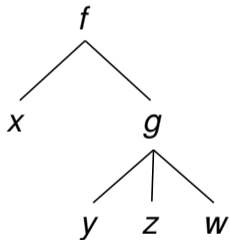
math. notation

$f(x\ g(y\ z\ w))$



Representations for Expressions

tree



prefix notation

$(f(x)(g(y)(z)(w)))$



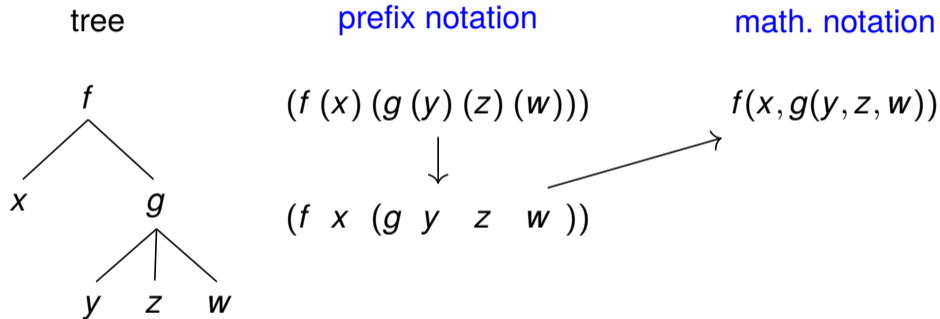
$(f x (g y z w))$

math. notation

$f(x, g(y, z, w))$



Representations for Expressions



Infix Notation

- ▶ 2-ary functions between parameters

Infix Notation

$f(x, y)$ $x f y$

- ▶ 2-ary functions between parameters

Infix Notation

$+(x, y)$ $x + y$

- ▶ 2-ary functions between parameters

Infix Notation

$$+(x, y) \quad x + y$$

▶ 2-ary functions between parameters

▶ **associativity**

$$(x f y) f z = x f (y f z)$$

Infix Notation

$$6 + 3 + 2 + 1$$

- ▶ 2-ary functions between parameters

- ▶ **associativity**

$$(x f y) f z = x f (y f z)$$

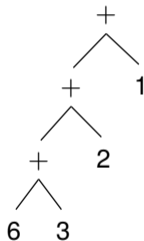
Infix Notation

$$6 + 3 + 2 + 1$$

▶ 2-ary functions between parameters

▶ **associativity**

$$(x f y) f z = x f (y f z)$$



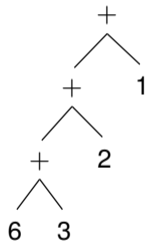
Infix Notation

$$6 + 3 + 2 + 1$$

▶ 2-ary functions between parameters

▶ **associativity**

$$(x f y) f z = x f (y f z)$$



left-associative

$$(x f y) f z$$

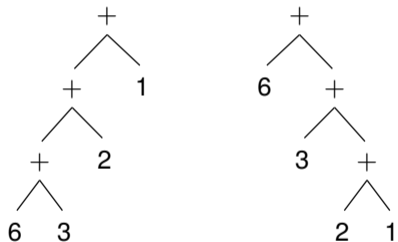
Infix Notation

$$6 + 3 + 2 + 1$$

- ▶ 2-ary functions between parameters

- ▶ **associativity**

$$(x f y) f z = x f (y f z)$$



left-associative

$$(x f y) f z$$

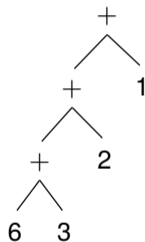
Infix Notation

$$6 + 3 + 2 + 1$$

► 2-ary functions between parameters

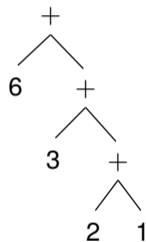
► **associativity**

$$(x f y) f z = x f (y f z)$$



left-associative

$$(x f y) f z$$



right-associative

$$x f (y f z)$$

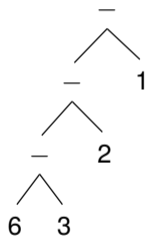
Infix Notation

- ▶ 2-ary functions between parameters

- ▶ **associativity**

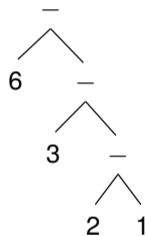
$$(x f y) f z = x f (y f z)$$

6 - 3 - 2 - 1



left-associative

$(x f y) f z$



right-associative

$x f (y f z)$

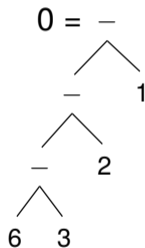
Infix Notation

- ▶ 2-ary functions between parameters

- ▶ **associativity**

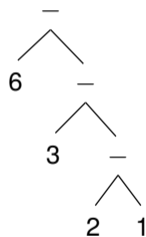
$$(x f y) f z = x f (y f z)$$

6 - 3 - 2 - 1



left-associative

$(x f y) f z$



right-associative

$x f (y f z)$

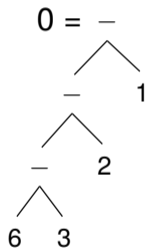
Infix Notation

▶ 2-ary functions between parameters

▶ **associativity**

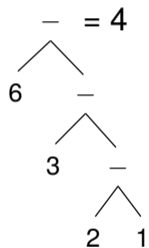
$$(x f y) f z = x f (y f z)$$

$$6 - 3 - 2 - 1$$



left-associative

$$(x f y) f z$$



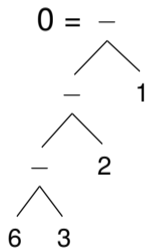
right-associative

$$x f (y f z)$$

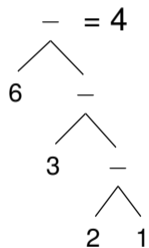
Infix Notation

- ▶ 2-ary functions between parameters
- ▶ **associativity**
 $(x f y) f z = x f (y f z)$
- ▶ **precedence**

6 - 3 - 2 - 1



left-associative
 $(x f y) f z$



right-associative
 $x f (y f z)$

Infix Notation

- ▶ 2-ary functions between parameters

- ▶ associativity

$$(x f y) f z = x f (y f z)$$

- ▶ precedence

Infix Notation

- ▶ 2-ary functions between parameters

- ▶ associativity

$$(x f y) f z = x f (y f z)$$

- ▶ precedence

$$2 + 3 \times 4$$

Infix Notation

- ▶ 2-ary functions between parameters

- ▶ **associativity**

$$(x f y) f z = x f (y f z)$$

- ▶ **precedence**

+ over ×

$$2 + 3 \times 4$$

$$(2 + 3) \times 4$$

Infix Notation

- ▶ 2-ary functions between parameters

- ▶ **associativity**

$$(x f y) f z = x f (y f z)$$

- ▶ **precedence**

+ over ×

× over +

$$2 + 3 \times 4$$

$$(2 + 3) \times 4 \quad 2 + (3 \times 4)$$

Assignments and Expressions

sequence of assignments

Assignments and Expressions

sequence of assignments

$$a \leftarrow (x + y) \times y$$

$$a \leftarrow (a \times a) + 1$$

$$b \leftarrow 5$$

$$c \leftarrow a - b$$

Assignments and Expressions

sequence of assignments

$$1: a \leftarrow (x + y) \times y$$

$$2: a \leftarrow (a \times a) + 1$$

$$3: b \leftarrow 5$$

$$4: c \leftarrow a - b$$

Assignments and Expressions

sequence of assignments

$$1: a \leftarrow (x + y) \times y$$

$$2: a \leftarrow (a \times a) + 1$$

$$3: b \leftarrow 5$$

$$4: c \leftarrow a - b$$

induced expressions

Assignments and Expressions

sequence of assignments

$$1: a \leftarrow (x + y) \times y$$

$$2: a \leftarrow (a \times a) + 1$$

$$3: b \leftarrow 5$$

$$4: c \leftarrow a - b$$

induced expressions

Assignments and Expressions

sequence of assignments

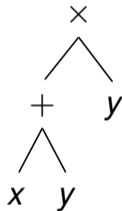
1: $a \leftarrow (x + y) \times y$

2: $a \leftarrow (a \times a) + 1$

3: $b \leftarrow 5$

4: $c \leftarrow a - b$

induced expressions



Assignments and Expressions

sequence of assignments

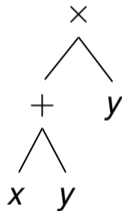
1: $a \leftarrow (x + y) \times y$

2: $a \leftarrow (a \times a) + 1$

3: $b \leftarrow 5$

4: $c \leftarrow a - b$

induced expressions



Assignments and Expressions

sequence of assignments

$$1: a \leftarrow (x + y) \times y$$

$$2: a \leftarrow (a \times a) + 1$$

$$3: b \leftarrow 5$$

$$4: c \leftarrow a - b$$

induced expressions

Assignments and Expressions

sequence of assignments

$$1: a \leftarrow (x + y) \times y$$

$$2: a \leftarrow (a \times a) + 1$$

$$3: b \leftarrow 5$$

$$4: c \leftarrow a - b$$

induced expressions

Assignments and Expressions

sequence of assignments

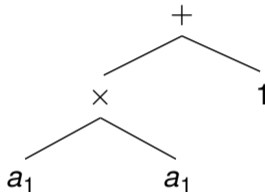
1: $a \leftarrow (x + y) \times y$

2: $a \leftarrow (a \times a) + 1$

3: $b \leftarrow 5$

4: $c \leftarrow a - b$

induced expressions



Assignments and Expressions

sequence of assignments

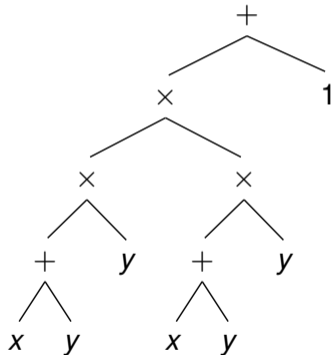
1: $a \leftarrow (x + y) \times y$

2: $a \leftarrow (a \times a) + 1$

3: $b \leftarrow 5$

4: $c \leftarrow a - b$

induced expressions



Assignments and Expressions

sequence of assignments

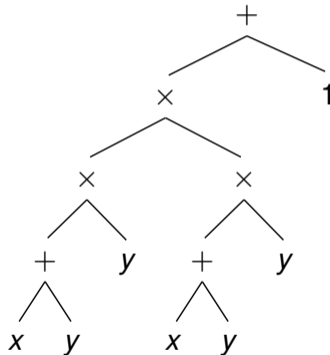
1: $a \leftarrow (x + y) \times y$

2: $a \leftarrow (a \times a) + 1$

3: $b \leftarrow 5$

4: $c \leftarrow a - b$

induced expressions



Assignments and Expressions

sequence of assignments

$$1: a \leftarrow (x + y) \times y$$

$$2: a \leftarrow (a \times a) + 1$$

$$3: b \leftarrow 5$$

$$4: c \leftarrow a - b$$

induced expressions

Assignments and Expressions

sequence of assignments

$$1: a \leftarrow (x + y) \times y$$

$$2: a \leftarrow (a \times a) + 1$$

$$3: b \leftarrow 5$$

$$4: c \leftarrow a - b$$

induced expressions

Assignments and Expressions

sequence of assignments

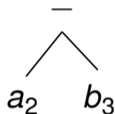
1: $a \leftarrow (x + y) \times y$

2: $a \leftarrow (a \times a) + 1$

3: $b \leftarrow 5$

4: $c \leftarrow a - b$

induced expressions



Assignments and Expressions

sequence of assignments

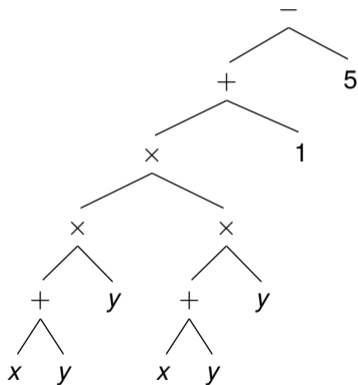
1: $a \leftarrow (x + y) \times y$

2: $a \leftarrow (a \times a) + 1$

3: $b \leftarrow 5$

4: $c \leftarrow a - b$

induced expressions



Assignments and Expressions

sequence of assignments

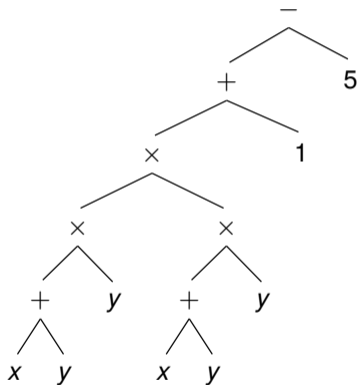
1: $a \leftarrow (x + y) \times y$

2: $a \leftarrow (a \times a) + 1$

3: $b \leftarrow 5$

4: $c \leftarrow a - b$

induced expressions



Assignments and Expressions

sequence of assignments

1: $a \leftarrow (x + y) \times y$

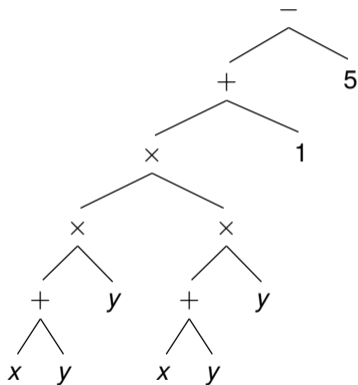
2: $a \leftarrow (a \times a) + 1$

3: $b \leftarrow 5$

4: $c \leftarrow a - b$

bottom-up construction of expression

induced expressions



Sources

- ▶ Computer Disk Icon (p. 2), Jonas Roßner, <https://openmoji.org>
- ▶ Cloud Icon (p. 2), Vanessa Boutzikoudi, <https://openmoji.org>
- ▶ Cantor's Pairing Function (p. 3), crh23, <https://commons.wikimedia.org>