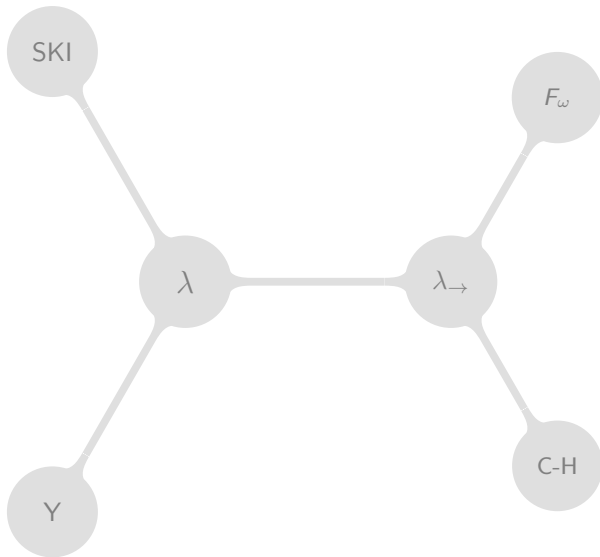


# Introduction to Lambda Calculus

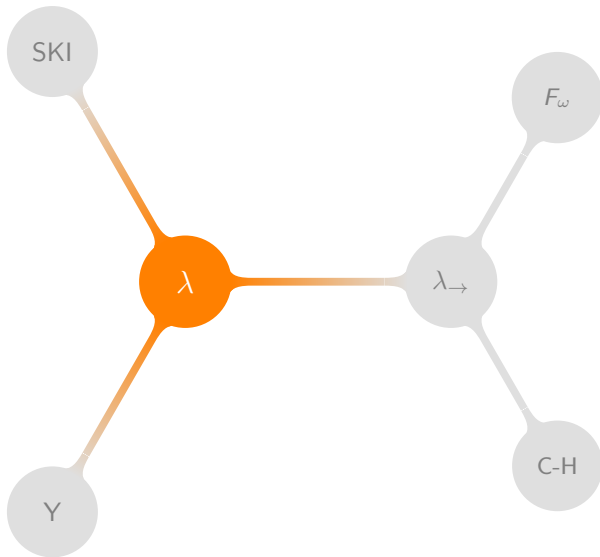
Maciek Makowski (@mmakowski)

12th October 2014

# The Plan



# Basic Lambda Calculus



# Syntax

$\langle term \rangle ::= x$	(variable)
$(\lambda x. \langle term \rangle)$	(abstraction)
$(\langle term \rangle \langle term \rangle)$	(application)

where  $x \in \mathbb{X}$  – the set of variables

# Syntax

$v_1$

# Syntax

$v_1$

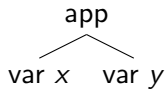
var  $v_1$

# Syntax

$x\ y$

# Syntax

$x\ y$





# Syntax

$\lambda a.b$

# Syntax

$\lambda a.b$

abs  $a$   
|  
var  $b$

# Syntax

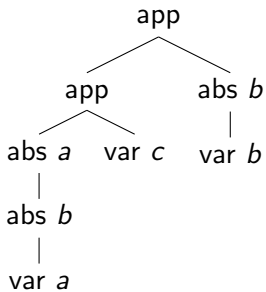
$(\lambda a. \lambda b. a) \ c \ (\lambda b. b)$

# Syntax

$\langle term \rangle ::= x$	(variable)
$(\lambda x. \langle term \rangle)$	(abstraction)
$(\langle term \rangle \langle term \rangle)$	(application)

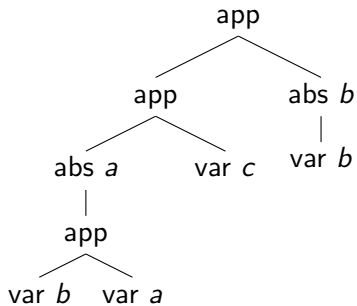
# Syntax

$(\lambda a. \lambda b. a) c (\lambda b. b)$



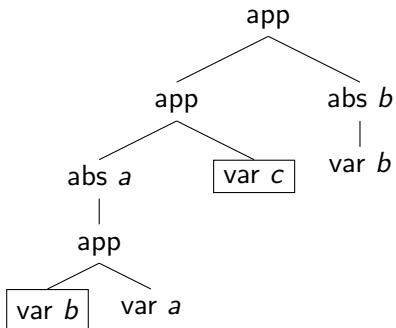
# Syntax

$(\lambda a. b \ a) \ c \ (\lambda b. b)$



# Syntax

$(\lambda a. \underline{b} \ a) \ \underline{c} \ (\lambda b. b)$



# Syntax

- ▶ terms: trees consisting of
  - ▶ variables
  - ▶ abstractions
  - ▶ applications
- ▶ variables are *bound* by abstraction; otherwise *free*



# Rewriting

$\alpha$ -conversion

$$(\lambda x.x\ y)\ (\lambda x.x) \longleftrightarrow_{\alpha} (\lambda a.a\ y)\ (\lambda b.b)$$

# Rewriting

$\beta$ -reduction

$$(\lambda x.M) N \longrightarrow_{\beta} M[x/N]$$

# Rewriting

$\beta$ -reduction

$$(\lambda x.M) N \longrightarrow_{\beta} M[x/N]$$

---

$$(\lambda x.x y) (\lambda z.z) \longrightarrow_{\beta} (\lambda z.z) y \longrightarrow_{\beta} y$$

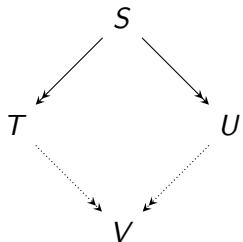
# Rewriting

$\beta$ -reduction

- ▶ *call-by-value*: start with innermost redex, do not reduce under abstraction
- ▶ *call-by-name*: start with outermost redex, do not reduce under abstraction

# Rewriting

Church-Rosser



# Semantics

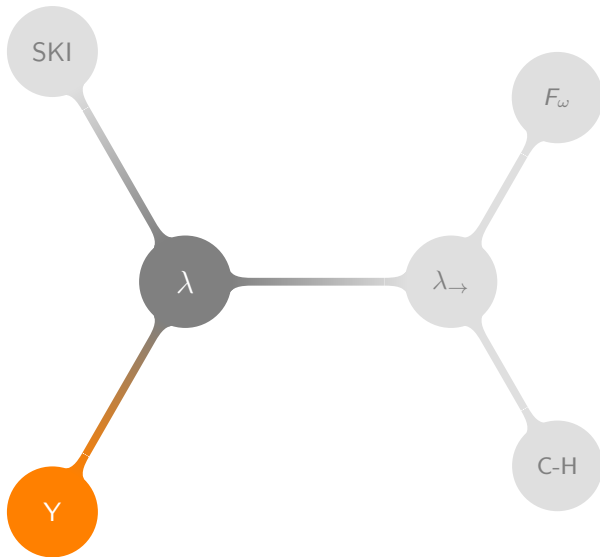
$$f(x) = a * x + b$$

# Semantics

$$f(x) = a * x + b$$

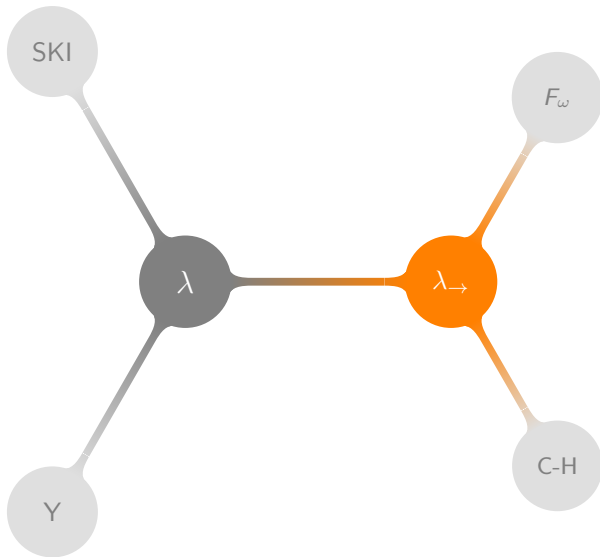
$$\lambda x. + (* a x) b$$

# Programming in Lambda Calculus

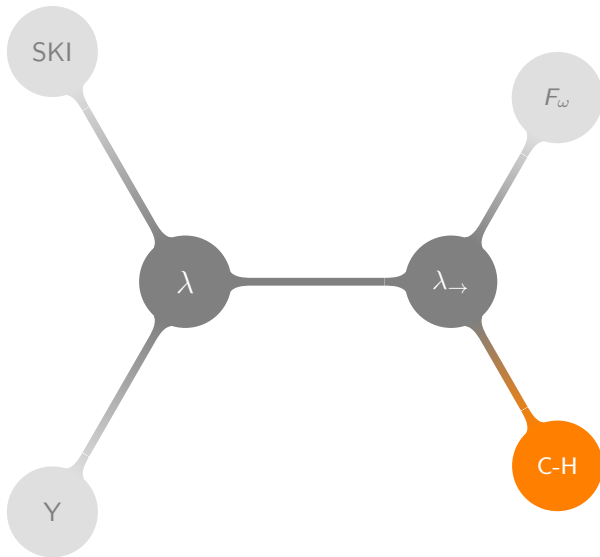




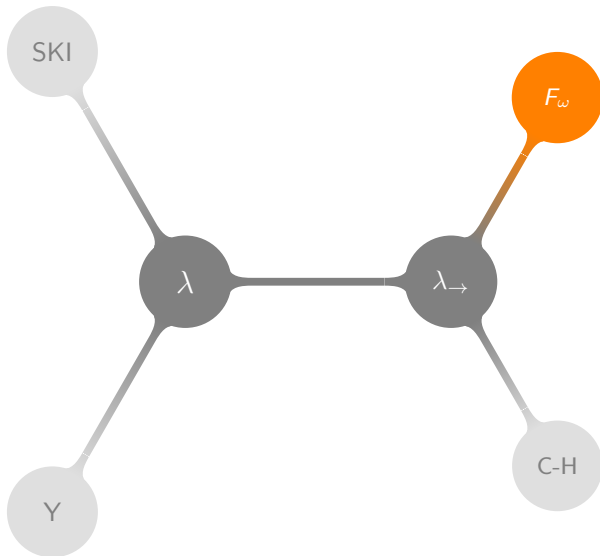
# Simple Types



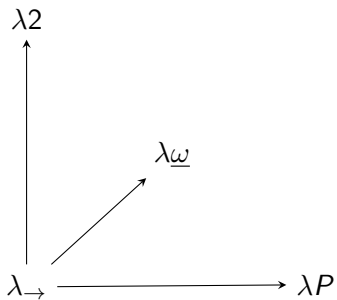
# Curry-Howard Correspondence



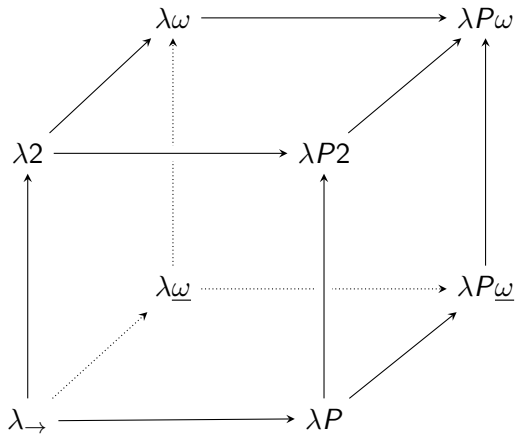
## More Types



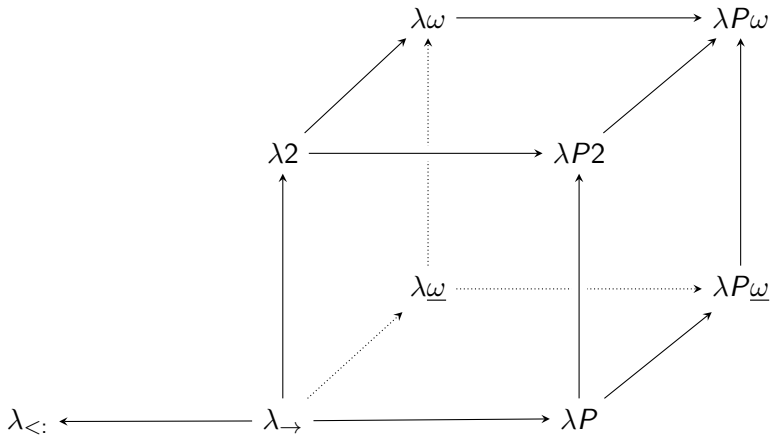
# The Lambda Cube



# The Lambda Cube



# Subtyping



# Subtyping

