Lambda-Calculus & Categories 6

9 November 2020

λ-calculus

an

introduction

I

λ-calculus an introduction!

a syntax for functions.

$$x \longmapsto x^2$$

function which associates the square of x to x.

$$\boxed{x \in Var}$$

1. we should start with a countable set Var of variables; every variable $x \in Var$ should be a term

2. we should be able to apply a function M to an argument N

$$\boxed{App\,(M,N)} \quad \boxed{MN}$$

argument

$$\boxed{f(x)}$$

function

3. we should be able given a term M of our syntax to construct a function

$$\boxed{x \longmapsto M}$$

where x may appear in M

This leads to the following grammar for $\underline{\lambda\text{-terms}}$:

$$\boxed{M ::= \; x \in Var \;\big|\; App(M,N) \;\big|\; \lambda x.\, M}$$

Example that we have in mind:

$$\boxed{I := \lambda x.\, x}$$ the identity function

$$\lambda f.\, \lambda x.\, App(f, x)$$

the function which takes two arguments f and x and applies f to x.

the starting point and spirit of Alonzo Church in the 1930's.

"A pure calculus of functions"

A subtlety: the two λ-terms

$$\lambda x. x \quad \text{and} \quad \lambda y. y$$

should be considered equal, in the same way as (for instance)

$$\int_0^1 t^2 \, dt = \int_0^1 x^2 \, dx.$$

We need to have careful treatment of "bound variables" in the syntax of the λ-calculus

Later in the course: species

combinatorial species

Speaking about notations:

λ

$$\vdash M : A$$

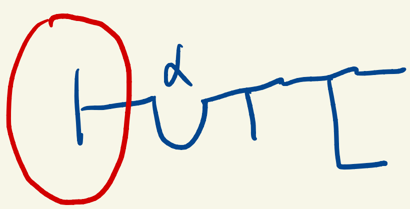
Frege's $\boxed{\text{ideography}}$ 1872

$\vdash \varphi$

a property φ which has been proved

$- \varphi$

a property φ



turnstyle notation

$$\boxed{\begin{array}{c} 2 + 5 \\ \\ V + W \end{array}}$$

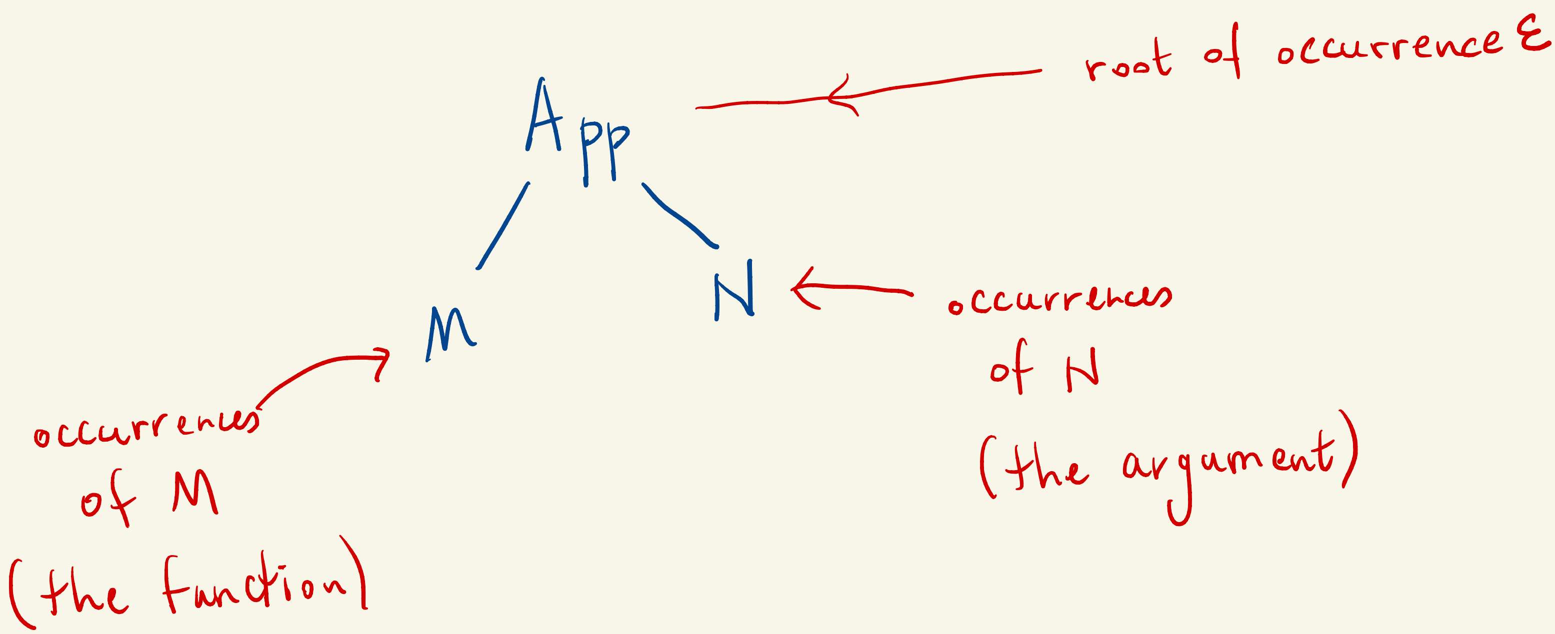
vector spaces

Occurrences of a λ-term.

<u>Def</u>. the set of occurrences of a λ-term is defined by induction:

① $$\mathrm{occ}(x) = \{\varepsilon\}$$
the empty word

② $$\mathrm{occ}(\mathrm{App}(M,N)) = \{\varepsilon\} \uplus \{\mathrm{fun}\cdot o \mid o \in \mathrm{occ}(M)\} \uplus \{\mathrm{arg}\cdot o \mid o \in \mathrm{occ}(N)\}$$

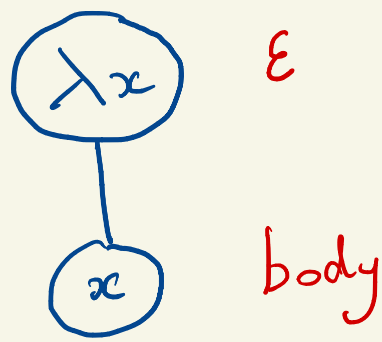


③ $$\mathrm{occ}(\lambda x.M) = \{\varepsilon\} \uplus \{\, \mathrm{body}\cdot o \mid o \in \mathrm{occ}(M) \,\}$$

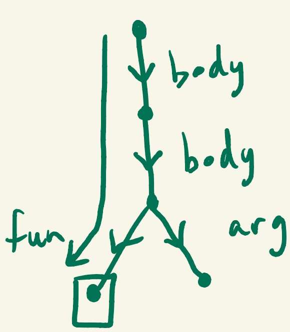
<u>Note</u>: every occurrence is a word on the alphabet $\{\mathrm{fun}, \mathrm{arg}, \mathrm{body}\}$ which indicates $\boxed{\text{a node}}$ in the λ-term M seen as a tree.

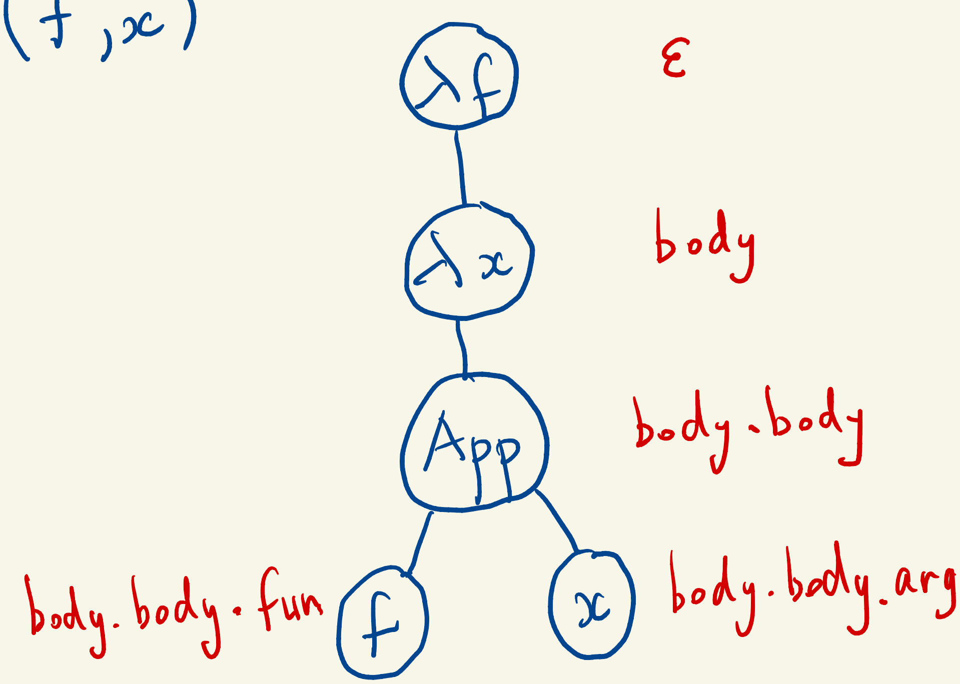
the path from the root to the node

<u>Example</u>: $\lambda x.x$



$\lambda f.\lambda x.\mathrm{App}(f,x)$





Free variables vs. bound variables

among the occurrences of a λ-term M

we may distinguish:

$\mathrm{occ}_\lambda(M)$ = the occurrence of a node labelled by λ

$\mathrm{occ}_{App}(M)$ = ______ labelled by App

$\mathrm{occ}_{var}(M)$ = ______ labelled by a variable

$$\mathrm{occ}(M) = \mathrm{occ}_{var}(M) \uplus \mathrm{occ}_{App}(M) \uplus \mathrm{occ}_\lambda(M)$$

Now we define for every λ-term M

a function:

$\underline{\mathrm{binder}}_M : \boxed{\mathrm{occ}_{var}(M) \to \mathrm{occ}_\lambda(M) \uplus \mathrm{Var}}$

which associates to every occurrence of a variable in M:

- the occurrence of the node λ which binds it (definition) when the occurrence of the variable is bound (definition)
- the variable x in Var when the occurrence of the variable is free in M.

$\mathrm{binder}(o) \qquad o \in occ_{var}(M)$

is defined by induction on M.

$$\boxed{\mathrm{binder}_M : occ_{var}(M) \to occ_{\lambda}(M) + Var}$$

(1) $\boxed{M = x}$ where ε is the occurrence of the variable

$$\text{binder}_x(\varepsilon) = x \in \text{Var}.$$

this means (see terminology above) that the occurrence of x at the root is free.

(since $\text{binder}_x(\varepsilon)$ is an element of Var)

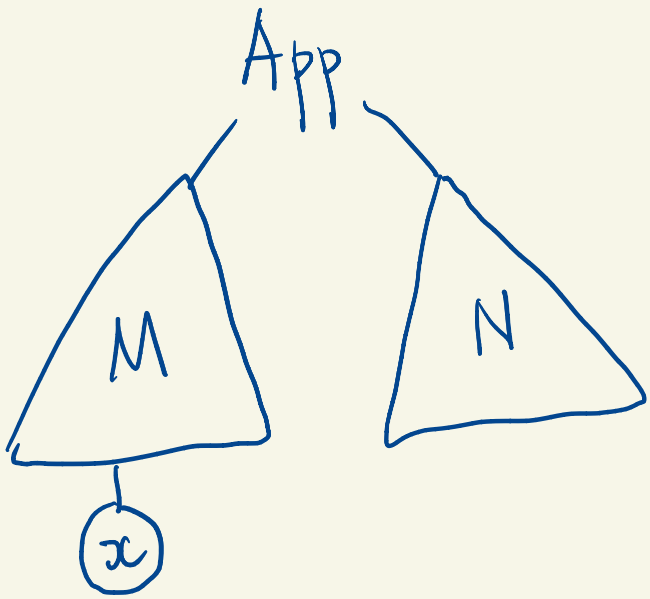
(2) $\boxed{P = \text{App}(M, N)}$

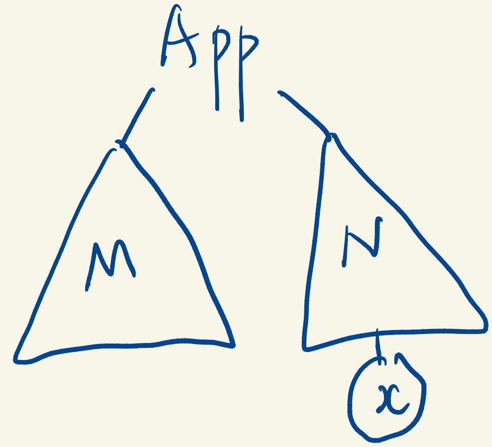
o is an occurrence of a <u>variable</u> in P

<u>two cases</u>: $\boxed{o = \text{fun} \cdot o'}$ or $\boxed{o = \text{arg} \cdot o'}$

when the occurrence of the variable is in M

when the occurrence is in N





in the first case:

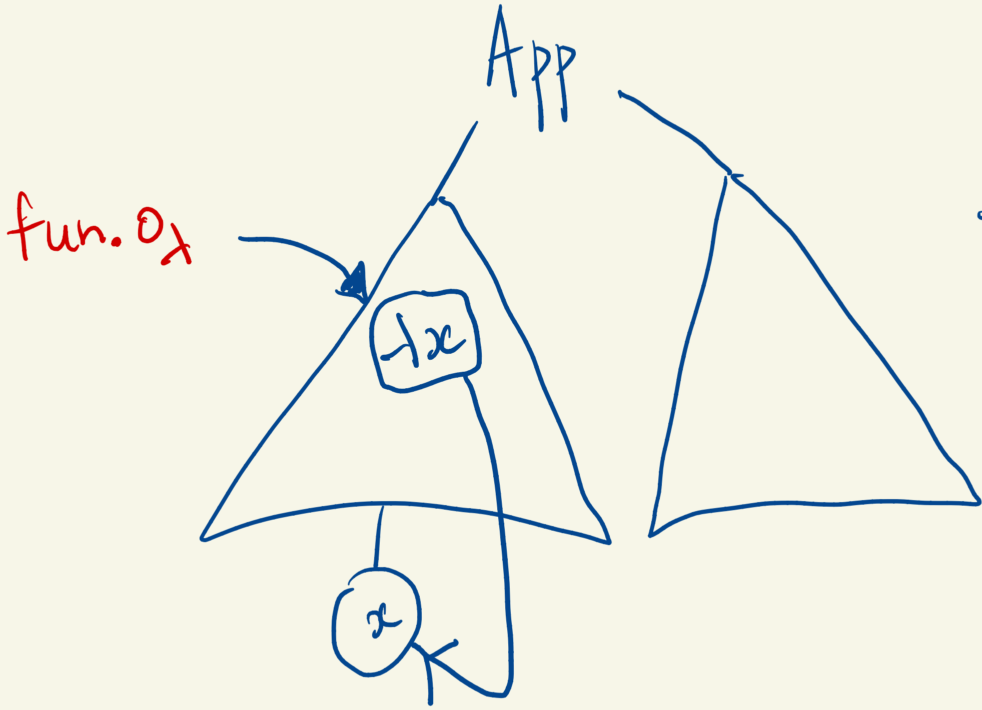
$$\mathrm{binder}_{\mathrm{App}(M,N)}(\mathrm{fun}\cdot o) = \mathrm{binder}_{M}(o) \in \mathrm{Var}$$

when the occurrence of the variable is free

$$\mathrm{binder}_{\mathrm{App}(M,N)}(\mathrm{fun}\cdot o) = \mathrm{fun}\cdot \mathrm{binder}_{M}(o)$$

when the occurrence of the variable is bound

note that $\mathrm{fun}\cdot\mathrm{binder}(o) \in \mathrm{occ}_{\lambda}(\mathrm{App}(M,N))$



$= App(M, N)$

the occurrence of this λx in $App(M,N)$

is $\boxed{fun.\ o_\lambda}$

where o_λ is the occurrence of λx in M

the second case works similarly:

$binder_{App(M,N)}(arg.\ o) = arg.\ binder_N(o)$

when o is bound (by a λ) in N

$binder_{App(M,N)}(arg.\ o) = binder_N(o) \in Var$

when o is free in N

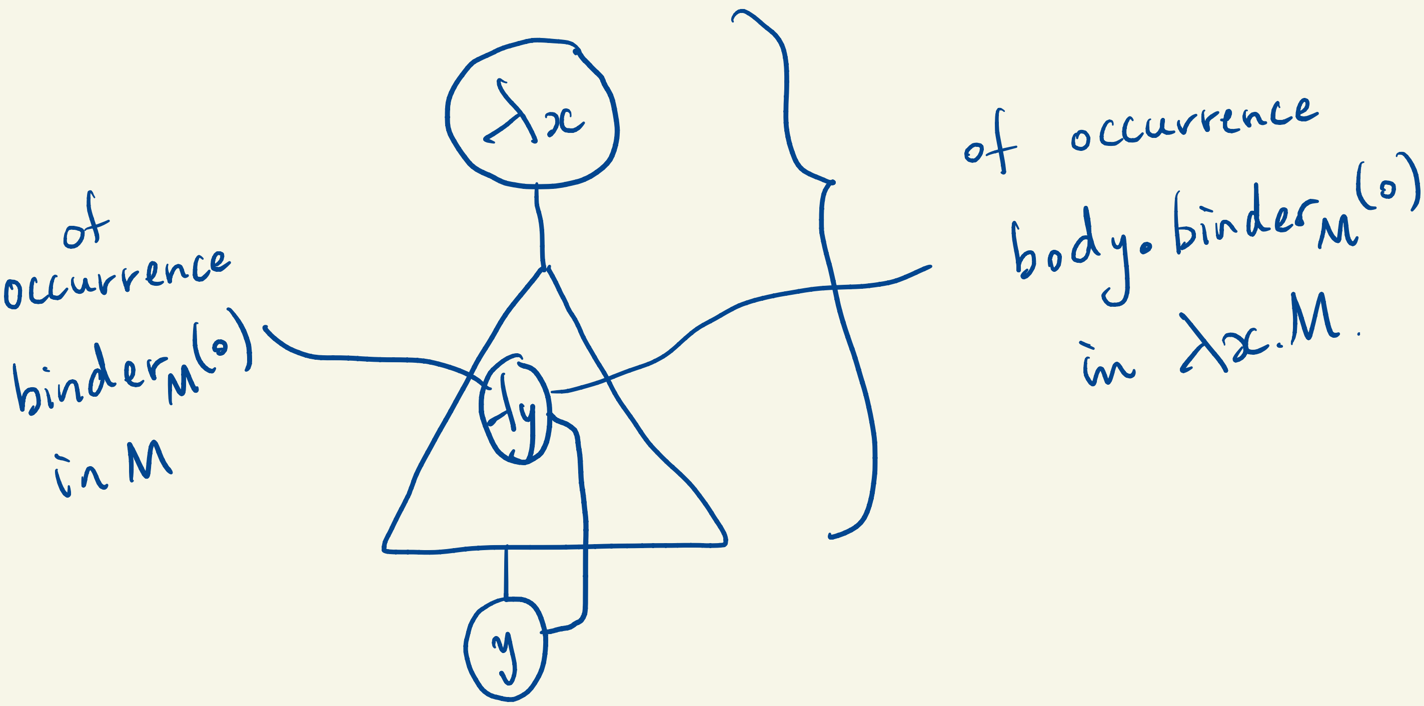
③ the interesting case:

$$\lambda x . M$$

given the occurrence o of a variable in M

ⓐ when the occurrence o is bound in M:

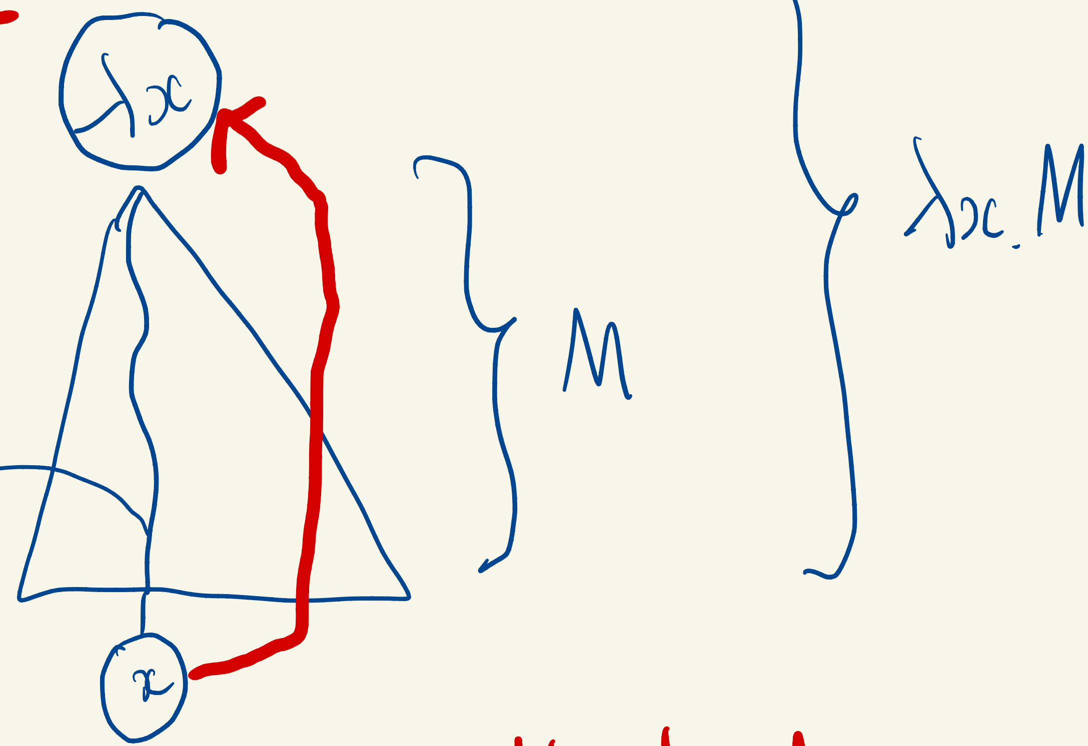
$$\mathrm{binder}_{\lambda x M}(\mathrm{body}.o) = \mathrm{body}.\mathrm{binder}_M(o)$$



ⓑ when $\mathrm{binder}_M(o) = x \in \mathrm{Var}$

something important happens:

$$\boxed{\text{binder}_{\lambda x.M}(\text{body}.o) = \varepsilon}$$



the occurrence o of a free variable x in the λ-term M

the binder of the occurrence of x is at the λ at the root

(c) when $\text{binder}_M(o) = y \in Var$

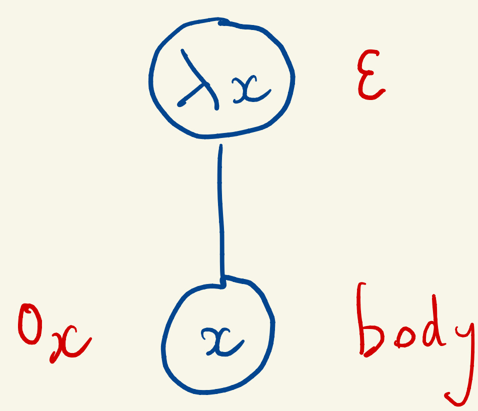
and $y \neq x$

then $\text{binder}_{\lambda x.M}(\text{body} \cdot o) = y \in Var$

the occurrence of the variable y remains free.

Examples:

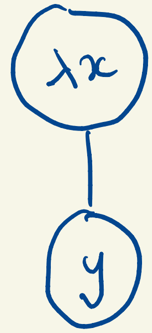
$\lambda x . x$



$\text{binder}_{\lambda x.x}(o_x) = \varepsilon$

$o_x = \text{body}$

$\lambda x. y$

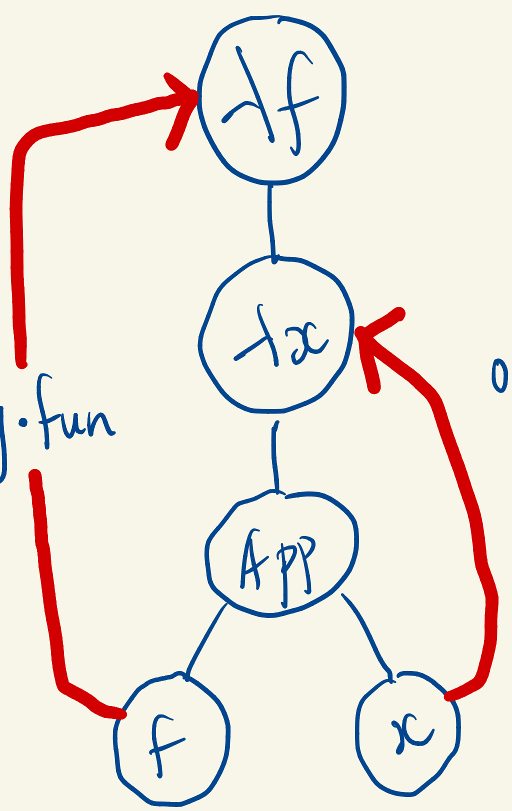


$o_y = \text{body}$

$\text{binder}_{\lambda x.y}(o_y) = y \in \text{Var}$

because y is a free variable.

$M = \lambda f. \lambda x. \text{App}(f, x)$



$o_f = \text{body.body.fun}$

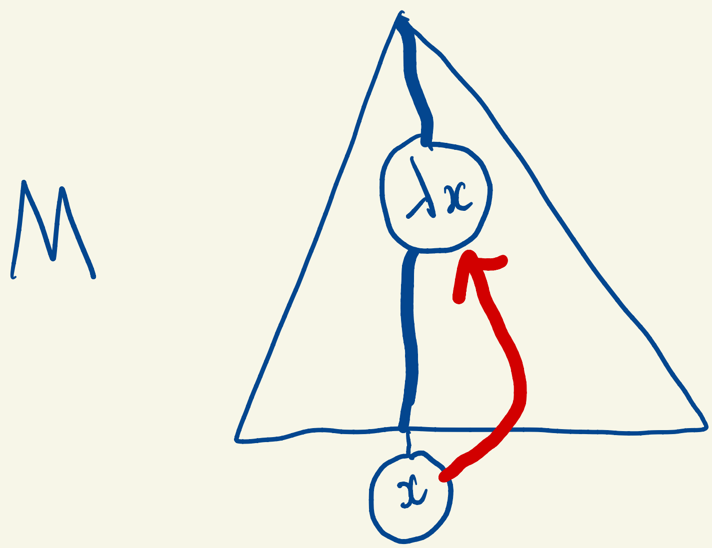
$o_x = \text{body.body.arg}$

$\text{binder}_M(o_f) = \varepsilon$

$\text{binder}_M(o_x) = \text{body}$

Remark: whenever the occurrence o_x of a variable x is bound in a λ-term M, then

1. $\mathrm{binder}_M(o_x)$ is the occurrence of a node labelled λx
2. the occurrence $\mathrm{binder}_M(o_x)$ is a prefix of the occurrence o_x.



Def. two λ-terms M and N are declared α-equivalent when

1. $\mathrm{occ}(M)$ and $\mathrm{occ}(N)$ are equal
2. binder_M and binder_N are equal

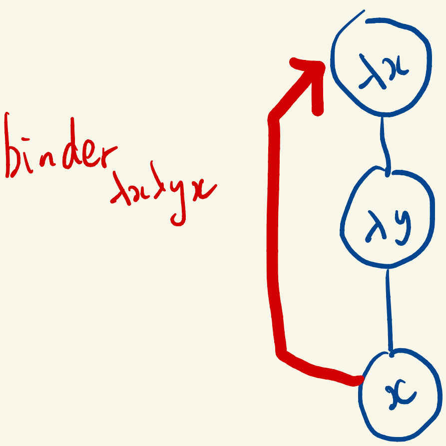
Traditionally, two λ-terms M and N are understood as "equal" when M and N are α-equivalent.

Notation: $\quad M \sim_\alpha N.$

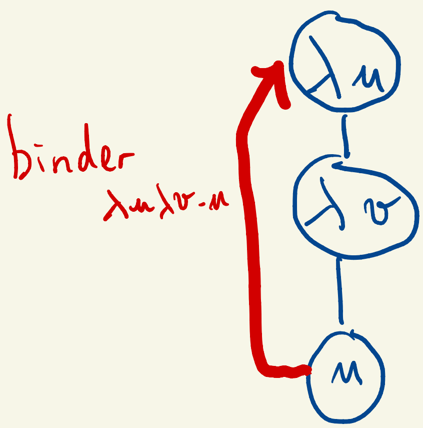
examples:

$$\lambda x.x \sim_\alpha \lambda y.y$$

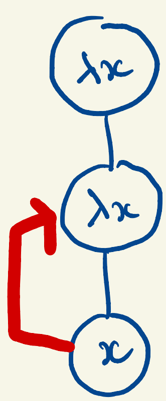
$$\lambda x.\lambda y.x \sim_\alpha \lambda u.\lambda v.u$$



$\sim_\alpha$



$\lambda x.\lambda x.x$



$\sim_\alpha$

$\lambda x.\lambda y.y$