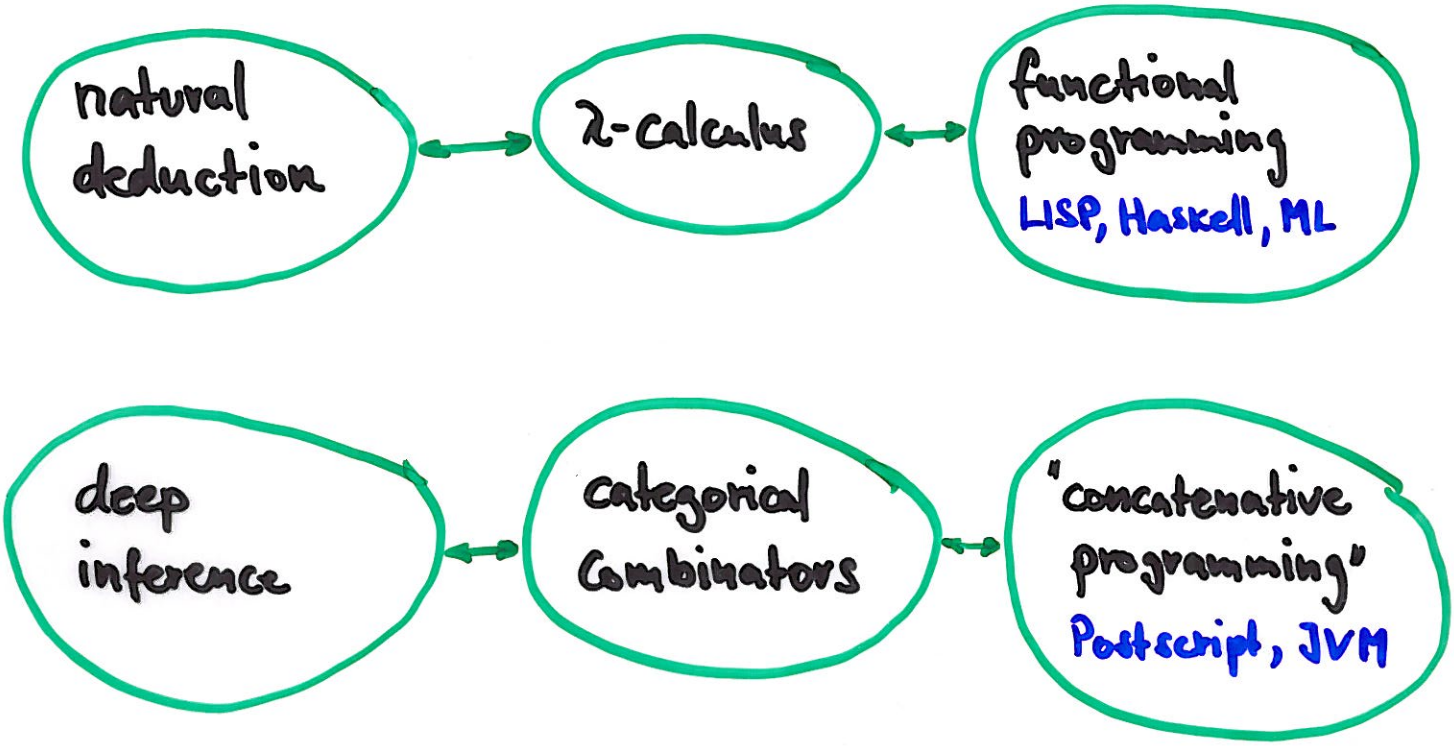
$\boxed{\text{GOAL}}$

SOLVE THE FOLLOWING EQUATION :

$$\frac{\text{natural deduction}}{\lambda\text{-calculus}} = \frac{\text{deep inference}}{\times}$$


natural deduction
λ-calculus
functional programming
LISP, Haskell, ML
deep inference
categorical combinators
"concatenative programming"
Postscript, JVM

natural deduction $\longleftrightarrow$ λ-calculus $\longleftrightarrow$ functional programming
LISP, Haskell, ML

deep inference $\longleftrightarrow$ categorical combinators $\longleftrightarrow$ "concatenative programming"
Postscript, JVM

apply $\wedge_E \dfrac{A \wedge B}{A}$ to $(a \wedge b) \wedge c$

the only possibility in natural deduction : $\wedge_E \dfrac{(a \wedge b) \wedge c}{a \wedge b}$

also possible in deep inference : $\wedge_E \dfrac{(a \wedge b) \wedge c}{a \wedge c}$

natural deduction

λ-calculus

functional programming
LISP, Haskell, ML

deep inference

categorical combinators

"concatenative programming"
Postscript, JVM

- developed by Curien in the 80ies to compile CAML into
- take the equational theory of the free cartesian closed category and orient them
- basis of the first explicit substitution calculus

natural deduction $\leftrightarrow$ λ-calculus $\leftrightarrow$ functional programming

LISP, Haskell, ML

deep inference $\leftrightarrow$ categorical combinators $\leftrightarrow$ "concatenative programming"

Postscript, JVM

functional	concatenative
let f x y z = (x+y)*z	let f = + . *
	"Concatenation"

A Deep Inference System for Intuitionistic Logic

$$\mathsf{c}\,\frac{A}{A \wedge A} \qquad \mathsf{i}\,\frac{B}{A \supset (B \wedge A)}$$

$$\mathsf{w_1}\,\frac{A \wedge B}{A} \qquad \mathsf{w_2}\,\frac{A \wedge B}{B} \qquad \mathsf{e}\,\frac{(A \supset B) \wedge A}{B}$$

Example Derivation

$$\mathsf{w_1}\,\frac{\mathsf{w_2}\,\dfrac{\mathsf{c}\,\dfrac{A \wedge B}{(A \wedge B) \wedge (A \wedge B)}}{B \wedge (A \wedge B)}}{B \wedge A}$$

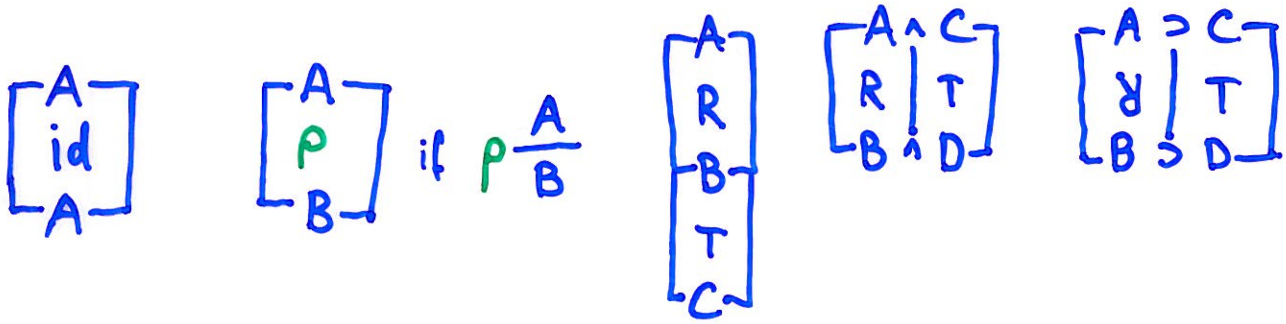
Proof Terms

$$R ::= id \mid \rho \in \{c, w_1, w_2, i, e\} \mid R \circ R \mid R \wedge R \mid R \supset R$$

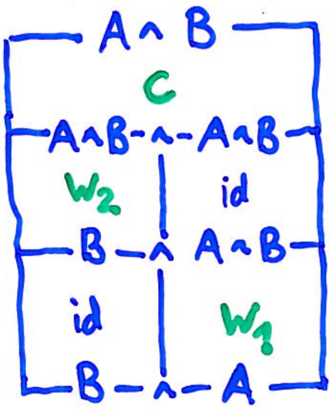
Example Term

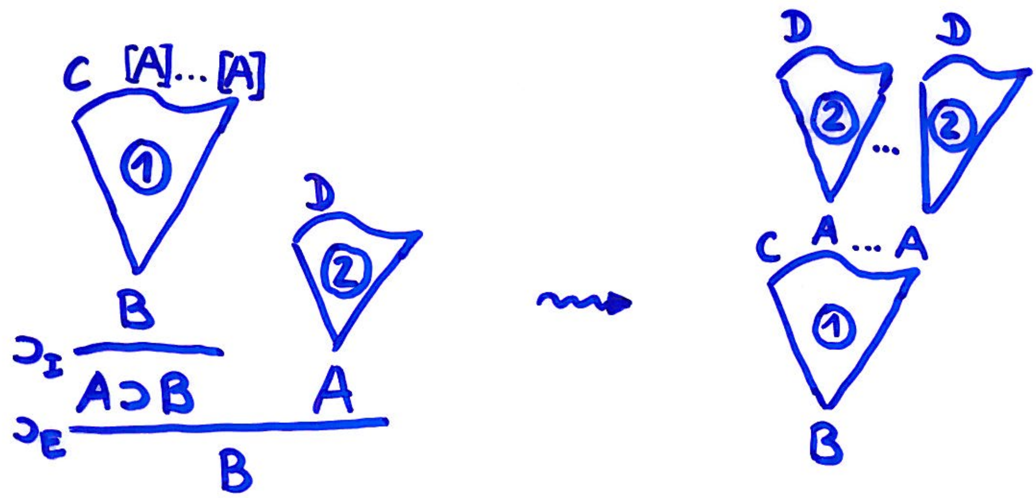
$$c \circ (w_2 \wedge id) \circ (id \wedge w_1)$$

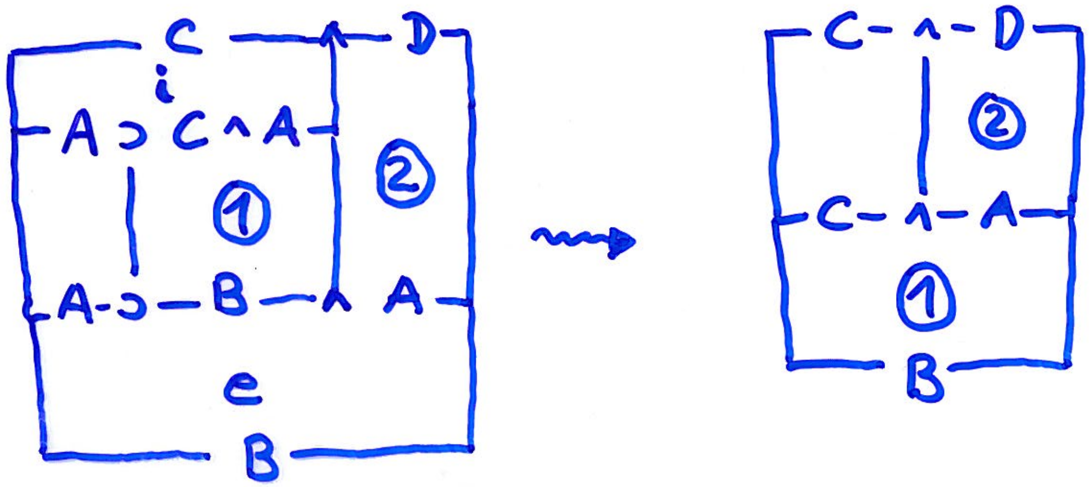
Typing Rules

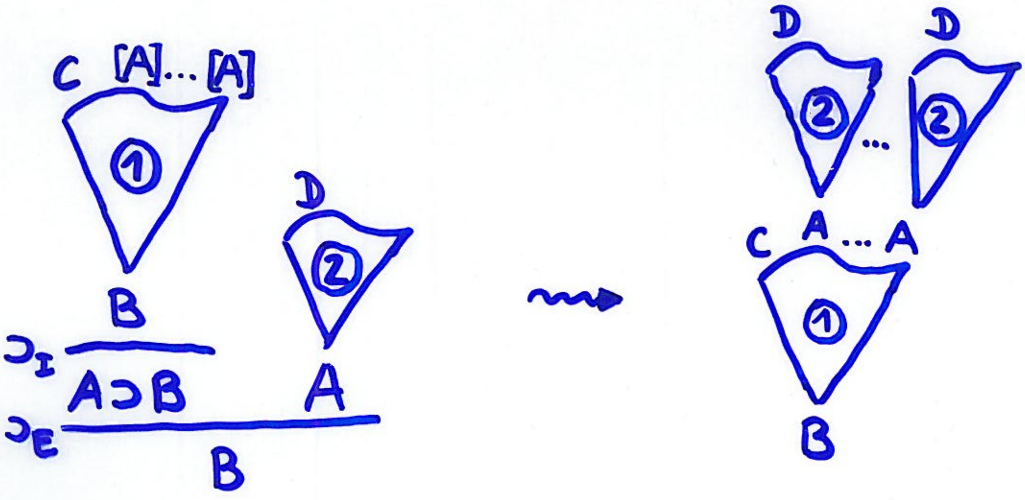


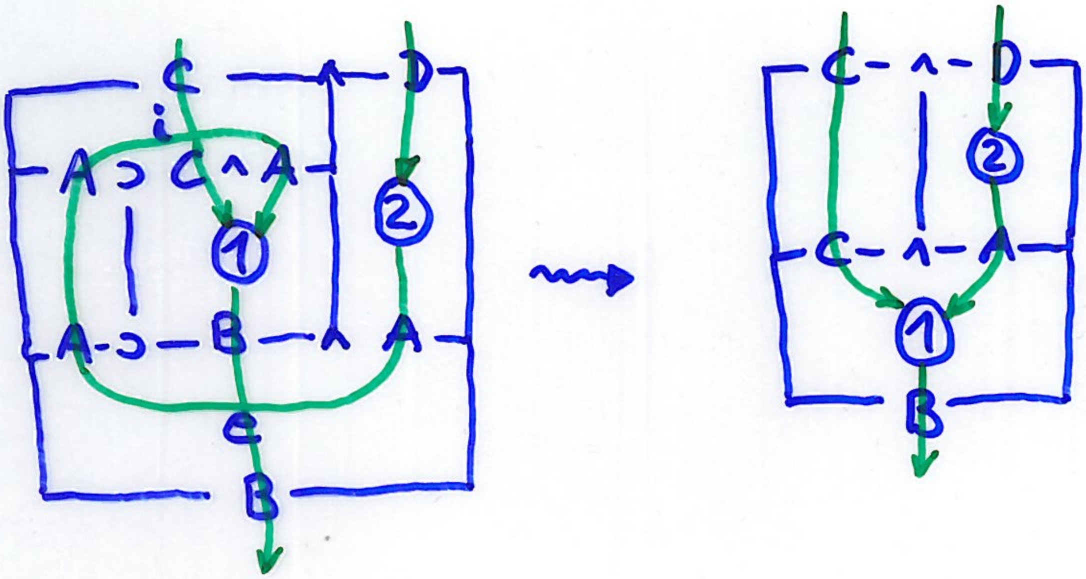
Typing Example



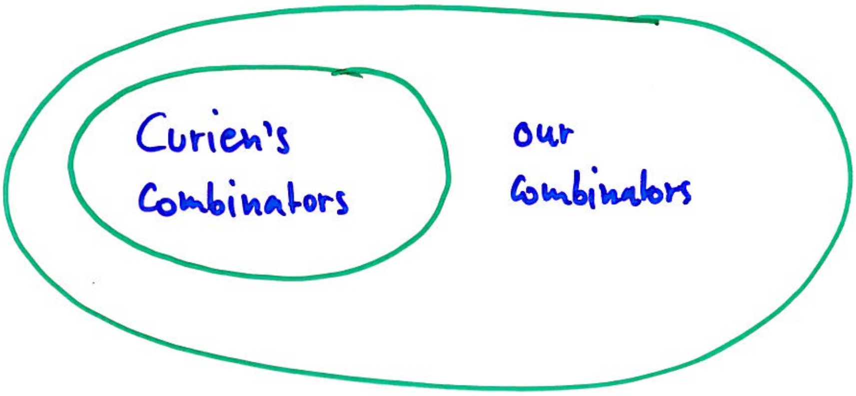

C [A]...[A]
1
B
⊃I
A ⊃ B
D
2
A
⊃E
B
D D
2 ... 2
C A... A
1
B


C — ∧ — D
i
A ⊃ C ∧ A
1
2
A — ⊃ — B — ∧ A
e
B
C — ∧ — D
2
C — ∧ — A
1
B


C [A]... [A]
1
B
D
2
A ⊃ B
A
B
D D
2 ... 2
C A ... A
1
B


C ∧ D
i
A ⊃ C ∧ A
1
2
A ⊃ B ∧ A
e
B
C ∧ D
2
C ∧ A
1
B

PROPERTIES OF THE COMBINATORS



	Curien	us
local confluence	NO	YES
confluence	NO	?
weak normalisation	YES	?
strong normalisation	NO	?