

# **Simple extensions**

# **Expressions with numbers**

## **Numerals**

- Terms and values (blackboard).
- New type **Nat** (blackboard).
- Typing rules (blackboard).
- Evaluation rules (blackboard).

## **Arithmetic expressions**

- Terms (blackboard).
- Typing rules (blackboard).
- Evaluation rules (blackboard).

## **Boolean expressions**

- Terms and values (blackboard).
- New type **Bool** (blackboard).
- Typing rules (blackboard).
- Evaluation rules (blackboard).

## Examples

- Find a derivation of the typing assertion

$$\vdash \mathbf{if\ isZero\ 0\ then\ 0\ else\ 2 + 3 : Nat}$$

(blackboard).

- Find a derivation of the typing assertion  $\vdash (\lambda x : \mathbf{Bool} . x) \mathbf{true} : \mathbf{Bool}$  (blackboard).
- Let  $t = \lambda x : \mathbf{Bool} . \mathbf{if\ } x \mathbf{\ then\ false\ else\ } f\ x$ . Find a derivation of the typing assertion  $f : \mathbf{Bool} \rightarrow \mathbf{Bool} \vdash t : \mathbf{Bool} \rightarrow \mathbf{Bool}$  (oneself).

## **Properties**

- Erasure (oneself).
- Permutation (oneself).
- Substitution (oneself).
- Uniqueness (blackboard).
- Progress (blackboard).
- Preservation (blackboard).
- Normalization (blackboard).

## **Product types**



## **Unit type**

- Terms and values (blackboard).
- New type **Unit** (blackboard).
- Typing rules (blackboard).
- Evaluation rules (blackboard).

## **Pair types**

- Terms and values (blackboard).
- New types of the form  $T \times U$  (blackboard).
- Typing rules (blackboard).
- Evaluation rules (blackboard).

## Examples

- For  $t = \text{fst } (4 - 1, \mathbf{if} \ \text{true} \ \mathbf{then} \ \text{false} \ \mathbf{else} \ \text{false})$ , find a type  $T$  such that  $\vdash t : T$  is derivable and an evaluation sequence reducing  $t$  to a value (blackboard).
- For  $t = (\lambda x : \mathbf{Nat} \times \mathbf{Nat} . \text{snd } x) (5 - 4, 4 + 5)$ , find a type  $T$  such that  $\vdash t : T$  is derivable and an evaluation sequence reducing  $t$  to a value (oneself).

## **Properties**

- Erasure (oneself).
- Permutation (oneself).
- Substitution (oneself).
- Uniqueness (blackboard).
- Progress (blackboard).
- Preservation (blackboard).
- Normalization (blackboard).

## **Sum types**

## **Bottom type**

- New type **Zero** (blackboard).

## **Variant types**

- Terms and values (blackboard).
- New types of the form  $T + U$  (blackboard).
- Typing rules (blackboard).
- Evaluation rules (blackboard).

## **Properties**

- Erasure (oneself).
- Permutation (oneself).
- Substitution (oneself).
- Ambiguity (oneself).
- Progress (blackboard).
- Preservation (blackboard).
- Normalization (blackboard).



## **Syntactic sugar**

## **Annotations**

- Terms (blackboard).
- Typing rules (blackboard).
- Evaluation rules (blackboard).
- Translation with correctness (blackboard).

## **Sequencing**

- Terms (blackboard).
- Typing rules (blackboard).
- Evaluation rules (blackboard).
- Translation with correctness (blackboard).

## **Wildcard pattern**

- Terms (blackboard).
- Typing rules (blackboard).
- Evaluation rules (blackboard).
- Translation with correctness (blackboard).

## **Generalized tuple types**

- Terms (blackboard).
- Typing rules (blackboard).
- Evaluation rules (blackboard).
- Translation with correctness (blackboard).

## **Generalized variant types**

- Terms (blackboard).
- Typing rules (blackboard).
- Evaluation rules (blackboard).
- Translation with correctness (blackboard).

## **Exercises**

- Exercise 11.9.1 (oneself).
- Redefine the syntax of and the rules concerning variant types in such a way that uniqueness holds (blackboard).

## **Let bindings**

- Terms (blackboard).
- Typing rules (blackboard).
- Evaluation rules (blackboard).
- Translation with correctness (blackboard).
- Exercise 11.5.2 (oneself).



# **Recursion**

## **Extensions**

- Terms (blackboard).
- Typing rules (blackboard).
- Evaluation rules (blackboard).

## Example

- For

$$t = \lambda f : \mathbf{Nat} \rightarrow \mathbf{Bool}. \lambda x : \mathbf{Nat}. \\ \mathbf{if} \ x == 0 \ \mathbf{then} \ \mathbf{true} \ \mathbf{else} \ \neg(f(x - 1)),$$

find a type  $T$  such that  $\vdash \mathbf{fix} \ t : T$  is derivable and find an evaluation sequence of  $\mathbf{let} \ e == \mathbf{fix} \ t \ \mathbf{in} \ e \ 3$  that ends with a value (blackboard).

## Exercises

- Exercise 11.11.1 (oneself).
- For

$$t = \lambda p : (\mathbf{Nat} \rightarrow \mathbf{Bool}) \times (\mathbf{Nat} \rightarrow \mathbf{Bool}).$$
$$(\lambda x : \mathbf{Nat} . \mathbf{if} \ x == 0 \ \mathbf{then} \ \mathbf{true} \ \mathbf{else} \ \mathbf{snd} \ p(x - 1),$$
$$\lambda x : \mathbf{Nat} . \mathbf{if} \ x == 0 \ \mathbf{then} \ \mathbf{false} \ \mathbf{else} \ \mathbf{fst} \ p(x - 1)),$$

find a type  $T$  such that  $\vdash \mathbf{fix} \ t : T$  is derivable and find an evaluation sequence of  $\mathbf{let} \ e == \mathbf{fix} \ t \ \mathbf{in} \ \mathbf{fst} \ e$  that ends with a value (oneself).

## **Properties**

- Erasure (oneself).
- Permutation (oneself).
- Substitution (oneself).
- Uniqueness (oneself).
- Progress (blackboard).
- Preservation (blackboard).
- Non-termination: occurrence of bottom in all types (blackboard).

## **Syntactic sugar**

- Terms (blackboard).
- Typing rules (oneself).
- Evaluation rules (oneself).
- Rewrite the example definition two slides back equivalently using the new syntactic sugar and redo the exercises (oneself).