

## nominal lisp

data Term where

$$V : \mathbb{A} \rightarrow \text{Term}$$
$$\mathsf{L} : \mathsf{List}(\mathsf{Term}) \rightarrow \mathsf{Term}$$
$$\mathbf{B} : \langle \mathbb{A} \rangle \text{Term} \rightarrow \text{Term}$$

Fundamental data types in the language:

- Symbol ( $\mathbb{A}$ )
- Pair
- Atom
- Binder

```
eval(env, symbol(s))      := lookup(env, s)
eval(env, pair(f, args)) :=
  let f_result = eval(env, f) in
  apply_or_special_form(f_result, args)
eval(env, atom(a))        := atom(a)
eval(env, binder(v, e))   := for v' # env,
                             binder(v', eval([v'/v]e))
```