$$\lambda x.\ ((\lambda y. f\ y)\ x)$$

$$\to_\beta \quad \lambda x.(\ f\ x\ )$$

$$\to_\eta \quad f$$

Lecture 12 Encodings & Rewrites

$$e \cong e'$$

Py

```
x = (1, 2)    # binary tuple
x = ()   # unit
x = (1, 2, True)  # triplet
```

Fb

$$e ::= x \mid i \mid b \mid \dots \mid \text{Let } x = e \text{ In } e \mid$$

$\mid \dots$

$\mid (e, e)$ ① tuple constructor

$\mid \text{Left } e \mid \text{Right } e$

② Left acc ③ Right Acc

$$e \Rrightarrow v$$

$$v ::= \ldots \mid (v, v)$$

$$[\text{Tuple}]\ \frac{e_1 \Rrightarrow v_1 \qquad e_2 \Rrightarrow v_2}{(e_1, e_2) \Rrightarrow (v_1, v_2)}$$

Introduced new syntax

$$[\text{Left}]\ \frac{e \Rrightarrow (v_1, v_2)}{\text{Left } e \Rrightarrow v_1}$$

Introduced new semantics

$$[\text{Right}]\ \frac{e \Rrightarrow (v_1, v_2)}{\text{Right } e \Rrightarrow v_2}$$

Tuples are primitive

Rewrites / Syntax Sugar

Py

```
for x in list:
    ...
```

Java

```
for (int i = 0 ; i < list.length(); i++) {
    x = list[i];
    ...
}
```

λ-calc

$$\mathrm{PAIR} \equiv_\delta \lambda x.\, \lambda y.\, \lambda f.\, f\ x\ y$$

$$\mathrm{LEFT} \equiv_\delta \lambda p.\, p\ (\lambda x.\, \lambda y.\, x)$$

$$\mathrm{RIGHT} \equiv_\delta$$

F^b

Desugaring

$$(e_1, e_2) \equiv_\delta ((\mathrm{Pair}\ e_1)\ e_2)$$

$$\mathrm{Pair} \equiv_\delta (\mathrm{Fun}\ x \to \mathrm{Fun}\ y \to \mathrm{Fun}\ f \to f\ x\ y)$$

$$\mathrm{Left}\ e \equiv_\delta (\mathrm{Left}\ e)$$

$$\mathrm{Left} \equiv_\delta (\mathrm{Fun}\ p \to p\ (\mathrm{Fun}\ x \to \mathrm{Fun}\ y \to x))$$

F^b_T $\qquad\qquad$ F^b

$$\underline{\underline{(e_1, e_2) \cong \mathrm{Pair}\ e_1\ e_2}}$$ (where Pair is defined above)

Definition of $\cong$

$$\underset{F^b_T}{\underset{\Uparrow}{(v_1, v_2)}} \cong \mathrm{Fun}\ f \to \underset{F^b}{\underset{\Uparrow}{f}}\ v_1\ v_2$$

$$\text{[Tuple]}_{F^{b}_{\tau}}\ \frac{e_1 \Rightarrow v_1 \qquad e_2 \Rightarrow v_2}{(e_1, e_2) \Rightarrow (v_1, v_2)}$$

$$\boxed{\text{[Tuple]}_{pb}\ \frac{e_1 \Rightarrow v_1 \qquad e_2 \Rightarrow v_2}{(e_1, e_2) \equiv_g \text{Pair } e_1\ e_2 \Rightarrow (v_1, v_2) \equiv_g \text{Fun } f \to f\ v_1\ v_2}}$$

We want to prove this is correct.

$$\frac{\dfrac{\star}{(\text{Fun } x \ldots)\ e_1 \Rightarrow \text{Fun } y \to \text{Fun } f \to f\ v_1\ y} \qquad e_2 \Rightarrow v_2 \qquad (\text{Fun } f \to f\ e_1\ y)[v_2/y] \Rightarrow \text{Fun } f \to f\ v_1\ v_2}{((\text{Fun } x \to \text{Fun } y \to \text{Fun } f \to f\,x\,y)\ e_1)\ e_2 \Rightarrow \text{Fun } f \to f\ v_1\ v_2}$$

1. LHS is a func 2. RHS is red to value 3. substitution

Rewrites & Hygeine

C Macro is not Hygienic

Rust Macro is Hygienic

Good

$$\mathrm{PAIR} \equiv_{\delta} (\lambda x . \lambda y . \lambda f . f\, x\, y)$$

$$\mathrm{PAIR}\; e_1\, e_2 \equiv_{\delta} (\quad \cdots \quad)\; e_1\; e_2$$

Bad

$$\mathrm{PAIR}'\; e_1\, e_2 \equiv_{\delta} \lambda f . f\, (e_1)\, (e_2)$$

$$e_1 \equiv f$$

$$\mathrm{PAIR}\; f\; e_2 \equiv_{\delta} (\lambda x . \lambda y . \lambda f . f\, x\, y)\; f\; e_2$$

$$\equiv_{\alpha} (\lambda x . \lambda y . \lambda g . g\, x\, y)\; f\; e_2$$

$$\equiv_{\beta} (\lambda y . \lambda g . g\, f\, y)\; e_2$$

$$\mathrm{PAIR}'\; f\; e_2 \equiv_{\delta} \lambda f . f\, f\, e_2$$

$$(\lambda t.\ \boxed{\text{IF TRUE}}\ (\text{LEFT } t)\ (\text{RIGHT } t))\ (\text{PAIR } a\, b)$$

$\Rightarrow a$

$\equiv_\delta$

$$\boxed{\text{IF TRUE}}$$

$$(\lambda b.\lambda t.\lambda f.\, b\, t\, f)\ (\lambda x.\lambda y.x)$$

$\equiv_\beta$ $\lambda t.\lambda f.((\lambda x.\lambda y.x)\ t\ f)$

$\equiv_\beta \ldots \equiv_\beta$ $\boxed{\lambda t.\lambda f.\ t}$

$$\boxed{(\lambda t.\lambda f.\ t)}\ (\text{LEFT } t)\ (\text{RIGHT } t)$$

$\equiv_\beta \ldots \equiv_\beta$ $(\text{LEFT } t)$

$$(\lambda t.\ \text{LEFT } t)\ (\text{Pair } a\, b)$$

$\equiv_\delta$ $(\lambda t.\ \text{LEFT } t)\ ((\lambda x.\lambda y.\lambda f.\ f\, x\, y)\ a\ b)$

$\equiv_\beta \ldots \equiv_\beta$ $(\lambda t.\ \text{LEFT } t)\ (\lambda f.\ f\, a\, b)$

$\equiv_\eta$ $\text{LEFT}\ (\lambda f.\ f\, a\, b)$

$\equiv_\delta$ $(\lambda p.\ p\ (\lambda x.\lambda y.x))\ (\lambda f.\ f\, a\, b)$

$\equiv_\beta$ $(\lambda f.\ f\, a\, b)\ (\lambda x.\lambda y.x)$

$$\equiv_\beta \quad (\lambda x.\lambda y.x)\ a\ b$$

$$\equiv_\beta \ldots \equiv_\beta \quad a$$