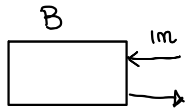
LCD (12/03/2024)

* VALUE PASSING CCS

pure (synchronisation) CCS

* Example

buffer



$\rightsquigarrow$ receives $n \in \mathbb{N}$

$\rightsquigarrow$ sends out $n+1$

$$B = \underline{in(x)}.\ \underline{B'(x)}$$

$$\underline{B'(x)} = \underline{\overline{out}(x+1).\ B}$$

behaviour

$$a(x).\ b(x).\ \overline{out}(x).0$$

input

$$\frac{}{a(x).\,P \xrightarrow{a(n)} P\{^n/_x\}} \qquad n \in \mathbb{N}$$

↖ any free occurrence of x is replaced by n

$$a(x).\ b(x).\ \overline{out}(x) \xrightarrow{a(x)} b(x).\ \overline{out}(x) \xrightarrow{b(x)} \overline{out}(x)$$

output

$$\frac{}{\bar{a}(e).\,P \xrightarrow{\bar{a}(n)} P} \qquad \text{if } e \text{ evaluates to } n$$

silent

$$\frac{}{\tau.P \xrightarrow{\tau} P}$$

constant

$$\frac{P\{^{n_1}/_{x_1}, \dots, ^{n_k}/_{x_k}\} \xrightarrow{\alpha} P'}{A(e_1, \dots, e_k) \xrightarrow{\alpha} P'} \qquad \begin{array}{l} e_1 \text{ evaluates to } n_1 \\ e_k \quad \text{"} \quad \text{"} \quad n_k \end{array}$$

$$A(x_1, \dots, x_k) \stackrel{def}{=} P$$

↖ conforming (possibly $x_1, \dots, x_k$)

* <u>how should I evaluate expressions with variables</u> ?

$$x+1 \xrightarrow{?}$$

I only execute programs which are closed (fully instantiated)

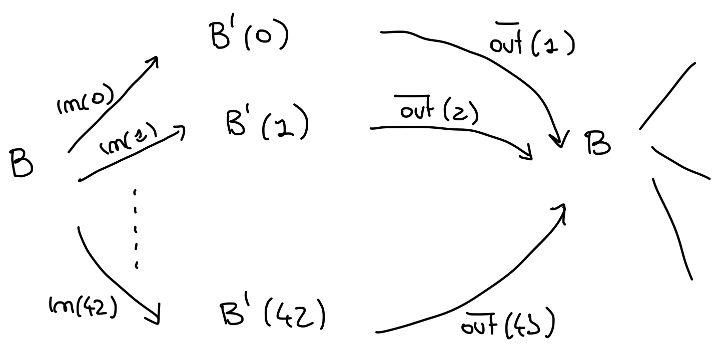
```
fact (n)                    fact (x + 1)
  if n = 0
    then 1                  fact (0)
  else                      fact (27)
    n * f(n-1)
```

if P has no free variables & $P \xrightarrow{\alpha} P'$ also P' has no free variables

<u>Example</u> :

$$B = in(x).\ B'(x)$$

$$B'(x) = \overline{out}(x+1).\ B$$



→ free variable

→ bound variable

→ reduction preserves closedness.

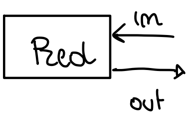
* conditionals

if b then P else Q

$$\frac{P \xrightarrow{\alpha} P'}{\text{if } b \text{ then } P \text{ else } Q \xrightarrow{\alpha} P'} \qquad \text{if } b \text{ evaluates to true}$$

$$\frac{Q \xrightarrow{\alpha} Q'}{\text{if } b \text{ then } P \text{ else } Q \xrightarrow{\alpha} Q'} \qquad \text{if } b \text{ evaluates to false}$$

<u>Exercise</u> : buffer



receives $n \in \mathbb{N}$

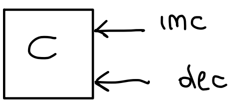
outputs $n-1$ if $n>0$

0 otherwise

$$Pred = in(x).\ P(x)$$

$$P(x) = \text{if } x=0 \text{ then } \overline{out}(0).\ Pred \text{ else } \overline{out}(x-1).\ Pred$$

<u>Exercise</u> :



counter

$$C(x) = inc.\ C(x+1) + \text{if } x>0 \text{ then } dec.\ C(x-1) \text{ else } \cancel{C(x)}\ 0$$

$$\frac{P \xrightarrow{\alpha} P'}{\text{if } b \text{ then } P \text{ else } Q \longrightarrow P'} \qquad \text{if } b \text{ true}$$

$$\begin{array}{llll}
\text{if } b_1 & \text{then} & chan_1(x).P_1 & + \\
\text{if } b_2 & \text{then} & chan_2(x).P_2 & + \\
\vdots & & & \\
\text{if } b_n & \text{then} & chan_K(x).P_K &
\end{array}$$

<u>CCS Value Passing</u>

- variables for values $\qquad x, y, \ldots \in Var$
- channels $\qquad a, b, c \in \mathcal{A}$
- process constants $\qquad K(x_1 \ldots, x_K) \in K$
- expressions
  - arithmetic $\qquad e ::= K \mid e + e \mid e * e \mid \ldots$
  - boolean $\qquad b ::= e = e \mid e \leq e \mid \neg b \mid b \wedge b$

Syntax

$$P, Q ::= K(e_1, \ldots, e_n) \mid a(x).P \mid \bar{a}(e).P \mid \tau.P \mid \text{if } b \text{ then } P$$

$$\uparrow$$

$$K(x_1, \ldots, x_n) \stackrel{def}{=} P \qquad \sum_{i \in I} P_i \mid P|Q \mid P \setminus L \mid P[f]$$

else ?

$$\text{if } b \text{ then } P \text{ else } Q \equiv \text{if } b \text{ then } P + \text{if } \neg b \text{ then } Q$$

$$\stackrel{?}{\equiv} \text{if } b \text{ then } P \mid \text{if } \neg b \text{ then } Q$$

$$\frac{}{\tau.P \xrightarrow{\tau} P}$$

$$\frac{}{a(x).P \xrightarrow{a(n)} P\{n/x\}} \qquad n \in \mathbb{N}$$

$$\frac{}{\bar{a}(e).P \xrightarrow{\bar{a}(n)} P} \qquad e \text{ evaluates to } n$$

$$\frac{P\{n_1/x_1, \dots, n_n/x_n\} \xrightarrow{\alpha} P'}{A(e_1, \dots, e_n) \xrightarrow{\alpha} P'} \qquad \begin{array}{l} e_i \text{ evaluates to } n_i \\ A(x_1, \dots, x_n) \overset{\text{def}}{=} P \end{array}$$

$$\frac{P \xrightarrow{\alpha} P'}{\text{if } b \text{ then } P \xrightarrow{\alpha} P'} \qquad \text{if } b \text{ evaluates to true}$$

$$\frac{P \xrightarrow{\bar{a}(n)} P' \qquad Q \xrightarrow{a(n)} Q'}{P \mid Q \xrightarrow{\tau} P' \mid Q'}$$

$$\boxed{\frac{}{a(x).P \xrightarrow{a(x)} P} \qquad \frac{P \xrightarrow{\bar{a}(n)} P' \qquad Q \xrightarrow{a(x)} Q'}{P \mid Q \xrightarrow{\tau} P' \mid Q'\{n/x\}}}$$

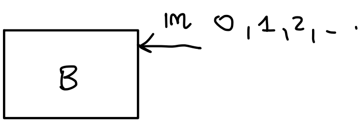
problems with variables

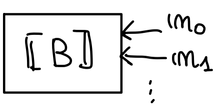
* Encoding CCS with value passing into pure CCS

$[\![\ ]\!] : \text{CCS-VP} \longrightarrow \text{CCS}$

$\mathcal{A}$ channels $\qquad \mathcal{A}' = \{ a_n \mid a \in \mathcal{A},\ n \in \mathbb{N} \}$

K constants $K(x_1, \dots, x_n) \qquad K' = \{ K_{n_1, \dots, n_n} \mid K(x_1, \dots, x_n) \in K \ \& \ n_1, \dots, n_n \in \mathbb{N} \}$





$\rightsquigarrow \quad \begin{array}{ll} B'(0) & B'_0 \\ B'(1) & B'_1 \\ \vdots & \vdots \end{array}$

$$[\![a(x).\, P]\!] = \sum_{m \in \mathbb{N}} a_m .\, [\![P\{^m/_x\}]\!]$$

$$[\![\bar{a}(e).\, P]\!] = \bar{a}_m .\, [\![P]\!] \qquad \text{if } e \text{ evaluates to } m$$

$$[\![\tau.\, P]\!] = \tau .\, [\![P]\!]$$

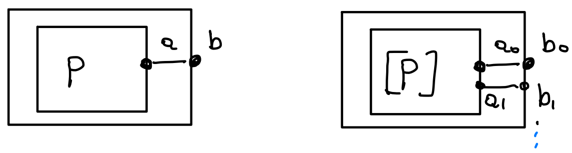
$$[\![\sum_{i \in I} P_i]\!] = \sum_{i \in I} [\![P_i]\!]$$

$$[\![P \mid Q]\!] = [\![P]\!] \mid [\![Q]\!]$$

$$[\![P \setminus L]\!] = [\![P]\!] \setminus \{ a_m \mid a \in L \}$$

$$[\![P[f]]\!] = [\![P]\!] [f']$$

$$f'(a_m) = f(a)_m$$



$$[\![\text{if } b \text{ then } P]\!] = \begin{cases} [\![P]\!] & b \text{ evaluates to true} \\ 0 & b \text{ evaluates to false} \end{cases}$$

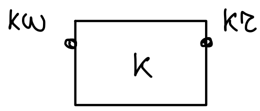
$$[\![K(e_1, \ldots, e_n)]\!] = K_{m_1, \ldots, m_n} \qquad e_i \text{ evaluates to } m_i$$

where $K_{m_1, \ldots, m_n} = [\![P \{^{m_1}/_{x_1}, \ldots, ^{m_n}/_{x_n}\}]\!]$ if $K(x_1, \ldots, x_n) = P$

* <u>Ex. Peterson</u>

variable $K \in \{1,2\}$

$$K(x) = \overline{kr}(x).\, K(x) + kw(y).\, K(y)$$



$$K1 = \overline{kr1}.\, K1 + \sum_{i \in \{1,2\}} kwi .\, Ki$$

$$= \overline{kr1}.\, K1 + kw1 .\, K1 + kw2 .\, K2$$

$K2 =$

MINI PROJECT: Write a compiler of CCS-VP into CCS

$x : [1 .. 42]$ integer intervals

$x : \{a, b, \dots c\}$ enumeration types

<u>Theorem</u>: Let $[\![\ ]\!] : CCS-VP \to CCS$ as above

Then for all CCS-VP programs P

(i) if $P \xrightarrow{\alpha} P'$ then $[\![P]\!] \xrightarrow{\hat{\alpha}} [\![P']\!]$

(ii) if $[\![P]\!] \xrightarrow{\hat{\alpha}} Q$ then there is P' s.t. $P \xrightarrow{\alpha} P'$ and $[\![P']\!] = Q$

where

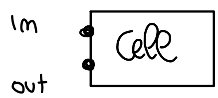
$$\hat{\alpha} = \begin{cases} a_n & \text{if } \alpha = a(n) \\ \overline{a_n} & \text{if } \alpha = \bar{a}(n) \\ \tau & \text{if } \alpha = \tau \end{cases}$$

<u>proof</u>

<u>EXERCISE</u>:

$$Cell \stackrel{def}{=} in(x).\ C(x)$$

$$C(x) \stackrel{def}{=} \overline{out}(x).\ Cell$$



① 2-place unordered buffer

$$B_2 = in(x).\ B_1(x)$$

$$B_1(x) = \overline{out}(x).\ B_2 + in(y).\ B_0(x,y)$$

$$B_0(x,y) = \overline{out}(x).\ B_1(y) + \overline{out}(y).\ B_1(x)$$

② FIFO BUFFER

$$F_2 = in(x).\ F_1(x)$$

$$F_1(x) = \overline{out}(x).\ F_2 + in(y).\ F_0(x,y)$$

$$F_0(x,y) = \overline{out}(x).\ F_1(y)$$