Traduction d'un programme fouine en fouine CPS

Dans la suite, on notera $\lambda x.\ e$ au lieu de $\texttt{fun}\ x \rightarrow e$, pour abréger les notations…

- $\llbracket n \rrbracket := \lambda k.\ (\mathit{fst}\ k)\ n$
- $\llbracket b \rrbracket := \lambda k.\ (\mathit{fst}\ k)\ b$
- $\llbracket () \rrbracket := \lambda k.\ (\mathit{fst}\ k)\ ()$
- $\llbracket x \rrbracket := \lambda k.\ (\mathit{fst}\ k)\ x$
- $\llbracket \lambda x.\ e \rrbracket := \lambda k.\ (\mathit{fst}\ k)\ (\lambda x.\ \llbracket e \rrbracket)$
- $\llbracket e_1\ e_2 \rrbracket = \lambda k.\ \llbracket e_2 \rrbracket\ (\lambda v.\ \llbracket e_1 \rrbracket\ (\lambda f.\ f\ v\ k, \mathit{snd}\ k), \mathit{snd}\ k)$
- $\llbracket e_1 \circledast e_2 \rrbracket := \lambda k.\ \llbracket e_2 \rrbracket\ (\lambda v_2.\ \llbracket e_1 \rrbracket\ (\lambda v_1.\ (\mathit{fst}\ k)\ (v_1 \circledast v_2), \mathit{snd}\ k), \mathit{snd}\ k)$
- $\llbracket \texttt{if}\ b\ \texttt{then}\ e_1\ \texttt{else}\ e_2 \rrbracket := \lambda k.\ \llbracket b \rrbracket\ (\lambda v.\ \texttt{if}\ v\ \texttt{then}\ \llbracket e_1 \rrbracket\ k\ \texttt{else}\ \llbracket e_2 \rrbracket\ k, \mathit{snd}\ k)$
- $\llbracket \circledast\ e \rrbracket := \lambda k.\ \llbracket e \rrbracket\ (\lambda v.\ (\mathit{fst}\ k)\ (\circledast\ v), \mathit{snd}\ k)$
- $\llbracket e_1\ \texttt{;}\ e_2 \rrbracket := \lambda k.\ \llbracket e_1 \rrbracket\ (\lambda\_.\ \llbracket e_2 \rrbracket\ k, \mathit{snd}\ k)$
- $\llbracket \mathrm{C}(e_1, \dots, e_n) \rrbracket := \lambda k.\ \llbracket e_n \rrbracket\ (\lambda v_n.\ \dots\ (\llbracket v_1 \rrbracket\ (\lambda v_1.\ (\mathit{fst}\ k)\ \ \mathrm{C}(v_1, \dots, v_n), \mathit{snd}\ k) \dots), \mathit{snd}\ k)$
- $$\begin{array}{l}\llbracket \texttt{while}\ b\ \texttt{do}\ e \rrbracket := \texttt{let rec}\ \mathit{boucle}\ k\ \texttt{=} \\ \qquad\qquad \llbracket b \rrbracket\ (\lambda v. \\ \qquad\qquad\quad \texttt{if}\ v\ \texttt{then}\ \llbracket e \rrbracket\ (\lambda\_.\ \mathit{boucle}\ k, \mathit{snd}\ k) \\ \qquad\qquad\quad \texttt{else}\ (\mathit{fst}\ k)\ () \\ \qquad\quad \texttt{in}\ \mathit{boucle}\end{array}$$
- $\llbracket \texttt{let rec}\ f\ \texttt{=}\ e\ \texttt{in}\ e' \rrbracket := \lambda k.\ \texttt{let rec}\ f\ \texttt{=}\ \llbracket e \rrbracket\ \texttt{in}\ \llbracket e' \rrbracket\ k$
- $$\begin{array}{l}\llbracket \texttt{for}\ i\ \texttt{=}\ e_1\ \texttt{to}\ e_2\ \texttt{do}\ e_3\ \texttt{done} \rrbracket := \lambda k.\ \llbracket e_1 \rrbracket\ (\lambda v_1.\ \llbracket e_2 \rrbracket\ (\lambda v_2. \\ \qquad\qquad \texttt{let rec}\ \mathit{boucle}\ i\ k\ \texttt{=} \\ \qquad\qquad\quad \texttt{if}\ i \leq v_2\ \texttt{then}\ \llbracket e_3 \rrbracket\ (\lambda\_\ .\ \mathit{boucle}\ (i+1)\ k, \mathit{snd}\ k) \\ \qquad\qquad\quad \texttt{else}\ (\mathit{fst}\ k)\ () \\ \qquad\qquad \texttt{in}\ \mathit{boucle}\ v_1))\end{array}$$
- $$\begin{array}{l}\llbracket \texttt{for}\ i\ \texttt{=}\ e_1\ \texttt{downto}\ e_2\ \texttt{do}\ e_3\ \texttt{done} \rrbracket := \lambda k.\ \llbracket e_1 \rrbracket\ (\lambda v_1.\ \llbracket e_2 \rrbracket\ (\lambda v_2. \\ \qquad\qquad \texttt{let rec}\ \mathit{boucle}\ i\ k\ \texttt{=} \\ \qquad\qquad\quad \texttt{if}\ i \geq v_1\ \texttt{then}\ \llbracket e_3 \rrbracket\ (\lambda\_\ .\ \mathit{boucle}\ (i-1)\ k, \mathit{snd}\ k) \\ \qquad\qquad\quad \texttt{else}\ (\mathit{fst}\ k)\ () \\ \qquad\qquad \texttt{in}\ \mathit{boucle}\ v_2))\end{array}$$
- $$\begin{array}{l}\llbracket \texttt{match}\ e\ \texttt{with}\ p_1\ \texttt{when}\ e'_1 \rightarrow e_1 \mid \cdots \mid p_n\ \texttt{when}\ e'_n \rightarrow e_n \rrbracket := \lambda k.\ \llbracket e \rrbracket\ (\lambda v. \\ \qquad\qquad \texttt{match}\ v\ \texttt{with} \\ \qquad\qquad \mid p_1\ \texttt{when}\ \llbracket e_{1'} \rrbracket\ (\mathit{id}, \mathit{snd}\ k) \rightarrow \llbracket e_1 \rrbracket\ k \\ \qquad\qquad \vdots\ \ \vdots\ \ \vdots\ \ \vdots\ \ \vdots\ \ \vdots\ \ \vdots\ \ \vdots\ \ \vdots\ \ \vdots \\ \qquad\qquad \mid p_n\ \texttt{when}\ \llbracket e_{n'} \rrbracket\ (\mathit{id}, \mathit{snd}\ k) \rightarrow \llbracket e_n \rrbracket\ k, \\ \qquad\quad \mathit{snd}\ k)\end{array}$$
- $$\begin{array}{l}\llbracket \texttt{try}\ e\ \texttt{with}\ p_1\ \texttt{when}\ e'_1 \rightarrow e_1 \mid \cdots \mid p_n\ \texttt{when}\ e'_n \rightarrow e_n \rrbracket := \lambda k.\ \llbracket e \rrbracket\ (\mathit{fst}\ k, \lambda v. \\ \qquad\qquad \texttt{match}\ v\ \texttt{with} \\ \qquad\qquad \mid p_1\ \texttt{when}\ \llbracket e_{1'} \rrbracket\ (\mathit{id}, \mathit{snd}\ k) \rightarrow \llbracket e_1 \rrbracket\ k \\ \qquad\qquad \vdots\ \ \vdots\ \ \vdots\ \ \vdots\ \ \vdots\ \ \vdots\ \ \vdots\ \ \vdots\ \ \vdots\ \ \vdots \\ \qquad\qquad \mid p_n\ \texttt{when}\ \llbracket e_{n'} \rrbracket\ (\mathit{id}, \mathit{snd}\ k) \rightarrow \llbracket e_n \rrbracket\ k \\ \qquad\qquad \mid \_ \rightarrow (\mathit{snd}\ k)\ v \\ \qquad\quad)\end{array}$$
- $\llbracket \texttt{raise}\ e \rrbracket := \lambda k.\ \llbracket e \rrbracket\ (\mathit{snd}\ k, \mathit{snd}\ k)$

On définit

- $\mathit{id} := \lambda x.\ x$
- $\mathit{fst} := \lambda (x, y).\ x$
- $\mathit{snd} := \lambda (x, y).\ y$