Assignment 1
Types

Sam Barrett, 1803086

October 20, 2020

1.

$$M ::= \lambda x.M|MM|\texttt{true}|\texttt{false}|\texttt{if}\ M\ \texttt{then}\ M\ \texttt{else}\ M|$$
$$\langle M, M\rangle|\texttt{spread}(M, M)|\texttt{left}(M)|\texttt{right}(M)$$

$$T ::= \mathbb{B}|T \to T|T \times T|T + T$$

(a) Extend call-by-value small-step operational semantics of the Curry-style Simply Typed λ-Calculus
Values, $V ::= \lambda x : T.M|\texttt{true}|\texttt{false}|\langle M, M\rangle|\texttt{left}(M)|\texttt{right}(M)$ Evaluation Contexts

$$C ::= \bullet|CM|VC|\texttt{if}\ C\ \texttt{then}\ M\ \texttt{else}\ M|\texttt{spread}(C, M)|\texttt{case}(C, M, M)$$

Rules:

$$\frac{}{(\lambda x : T.M)V \to_v M[x \backslash V]}\ \beta$$

$$\frac{}{\texttt{if true then}\ M\ \texttt{else}\ N \to_v M}\ \text{ITET}$$

$$\frac{}{\texttt{if false then}\ M\ \texttt{else}\ N \to_v N}\ \text{ITEF}$$

$$\frac{M \to_v N}{C[M] \to_v C[N]}\ \texttt{CTX}_C$$

$$\frac{M \to_v \langle P, Q\rangle}{\texttt{spread}(M, N) \to_v NPQ}\ \text{SP}$$

$$\frac{M \to_v \texttt{left}(N)}{\texttt{case}(M, P, Q) \to_v NP}\ \text{CL}$$

$$\frac{M \to_v \texttt{right}(N)}{\texttt{case}(M, P, Q) \to_v NQ}\ \text{CR}$$

(b) Prove: **Note: unless stated otherwise Π_n stands for the RHS of the bottom of the previous proof tree**

$$\texttt{spread}((\lambda x.\langle x, x\rangle)\texttt{true}, \lambda y.\lambda z.\texttt{if}\ \ y\ \ \texttt{then}\ \ \texttt{left}(z)\ \ \texttt{else}\ \ \texttt{right}(z)) \to_v^* \texttt{left(true)}$$

$$\frac{\dfrac{\langle\texttt{true,true}\rangle}{(\lambda x.\langle x, x\rangle)\texttt{true}}\ \beta}{\Pi_1}\ \texttt{spread}(\bullet, M) \qquad (1)$$

Where Π_1 is defined as:

$$\texttt{spread}((\lambda x.\langle x, x\rangle)\texttt{true}, \lambda y.\lambda z.\texttt{if}\ \ y\ \ \texttt{then}\ \ \texttt{left}(z)\ \ \texttt{else}\ \ \texttt{right}(z)) \to_v \texttt{spread}(\langle\texttt{true,true}\rangle, \texttt{if}\ \ y\ \ \texttt{then}\ \ \texttt{left}(z)\ \ \texttt{else}\ \ \texttt{right}(z))$$

$$\frac{}{\Pi_1 \to_v (\lambda y.\lambda z.\texttt{if}\ \ y\ \ \texttt{then}\ \ \texttt{left}(z)\ \ \texttt{else}\ \ \texttt{right}(z))\texttt{true true}}\ \text{SP} \qquad (2)$$

$$\frac{}{\Pi_2 \to_v \lambda z.\texttt{if true then left}(z)\ \ \texttt{else right}(z)}\ \beta \qquad (3)$$

$$\frac{}{\Pi_3 \to_v \texttt{if true then left(true) else right(true)}}\ \beta \qquad (4)$$

$$\frac{}{\Pi_4 \to_v \texttt{left(true)}}\ \text{ITET} \qquad (5)$$

(c) Typing Rules:

$$\frac{}{\Gamma, x : T \vdash x : T}\ \text{VAR}$$

$$\frac{\Gamma, x : T \vdash M : U}{\Gamma \vdash \lambda x : T.M : T \to U}\ \text{ABS}$$

$$\frac{\Gamma \vdash M : T \to U \qquad \Gamma \vdash N : T}{\Gamma \vdash MN : U}\ \text{APP}$$

$$\frac{}{\Gamma \vdash \texttt{true} : \mathbb{B}}\ \text{T}$$

$$\frac{}{\Gamma \vdash \texttt{false} : \mathbb{B}}\ \text{F}$$

$$\frac{\Gamma \vdash M : \mathbb{B} \qquad \Gamma \vdash N : T \qquad \Gamma \vdash P : T}{\Gamma \vdash \texttt{if}\ \ M\ \ \texttt{then}\ \ N\ \ \texttt{else}\ \ P : T}\ \text{ITE}$$

$$\frac{\Gamma \vdash M : T \qquad \Gamma \vdash M : U}{\Gamma \vdash \langle M, N\rangle : T \times U}\ \text{PAIR}$$

$$\frac{\Gamma \vdash M : T \times U \qquad \Gamma \vdash N : T \to U \to Z}{\Gamma \vdash \texttt{spread}(M, N) : Z} \text{ SP}$$

$$\frac{\Gamma \vdash M : T}{\Gamma \vdash \texttt{left}(M) : T + U} \text{ L}$$

$$\frac{\Gamma \vdash M : U}{\Gamma \vdash \texttt{right}(M) : T + U} \text{ R}$$

$$\frac{\Gamma \vdash M : T \qquad \Gamma \vdash P : T \to Z}{\Gamma \vdash \texttt{case}(\texttt{left}(M), P, Q) : Z} \text{ CASEL}$$

$$\frac{\Gamma \vdash M : U \qquad \Gamma \vdash Q : U \to Z}{\Gamma \vdash \texttt{case}(\texttt{right}(M), P, Q) : Z} \text{ CASER}$$

(d) Prove:

$$\texttt{spread}((\lambda x.\langle x, x\rangle)\texttt{true}, \lambda y.\lambda z.\texttt{if}\ \ y\ \ \texttt{then}\ \ \texttt{left}(z)\ \ \texttt{else}\ \ \texttt{right}(z))$$

has type $\mathbb{B} + \mathbb{B}$

$$\frac{\overline{\texttt{true} : \mathbb{B}} \text{ T}}{\texttt{left}(\texttt{true}) : \mathbb{B} + U} \text{ L}$$

Where U is of **any** type, $\therefore$ we can say $U \mapsto \mathbb{B}$ giving us $\texttt{left}(\texttt{true}) : \mathbb{B} + \mathbb{B}$

2. Is:

$$(\lambda x.\lambda y.((xy)(x\texttt{true})$$

typeable within the Simply Typed λ-Calculus?

Wand's Algorithm:

Note: where '$C = C$', C has not changed since the previous line

$$C = \emptyset, G = \langle \cdot; M_0; \alpha_0 \rangle \tag{1}$$

$$C = \{\alpha_0 = \alpha_1 \to \alpha_2\}, G = \langle x : \alpha_1; \lambda y.((xy)(x\texttt{true})); \alpha_2 \rangle \tag{2}$$

$$C = \{\alpha_0 = \alpha_1 \to \alpha_2, \alpha_2 = \alpha_3 \to \alpha_4\}, G = \langle x : \alpha_1, y : \alpha_3; (xy)(x\texttt{true}); \alpha_4 \rangle \tag{3}$$

$$C = C, G = \{\langle x : \alpha_1, y : \alpha_3; xy; \alpha_5 \to \alpha_4 \rangle, \langle x : \alpha_1, y : \alpha_3; x\texttt{true}; \alpha_5 \rangle\} \tag{4}$$

$$\begin{aligned} C &= C, \\ G &= \{\langle x:\alpha_1, y:\alpha_3; x; \alpha_6 \to \alpha_5 \to \alpha_4\rangle, \langle x:\alpha_1, y:\alpha_3; y; \alpha_6\rangle, \langle x:\alpha_1, y:\alpha_3; x\texttt{true}; \alpha_5\rangle\} \end{aligned} \tag{5}$$

$$C = C, G = \{\langle x:\alpha_1, y:\alpha_3; x; \alpha_6 \to \alpha_5 \to \alpha_4\rangle, \langle x:\alpha_1, y:\alpha_3; y; \alpha_6\rangle, \\ \langle x:\alpha_1, y:\alpha_3; x; \alpha_7 \to \alpha_5\rangle, \langle x:\alpha_1, y:\alpha_3; \texttt{true}; \alpha_7\rangle\} \tag{6}$$

$$\begin{aligned} C &= \{\alpha_0 = \alpha_1 \to \alpha_2, \alpha_2 = \alpha_3 \to \alpha_4, \alpha_6 \to \alpha_5 \to \alpha_4 = \alpha_1\}, \\ G &= \{\langle x:\alpha_1, y:\alpha_3; y; \alpha_6\rangle, \langle x:\alpha_1, y:\alpha_3; x; \alpha_7 \to \alpha_5\rangle, \langle x:\alpha_1, y:\alpha_3; \texttt{true}; \alpha_7\rangle\} \end{aligned} \tag{7}$$

$$C = \{\alpha_0 = \alpha_1 \to \alpha_2, \alpha_2 = \alpha_3 \to \alpha_4, \alpha_6 \to \alpha_5 \to \alpha_4 = \alpha_1, \alpha_6 = \alpha_3\}, \\ G = \{\langle x:\alpha_1, y:\alpha_3; x; \alpha_7 \to \alpha_5\rangle, \langle x:\alpha_1, y:\alpha_3; \texttt{true}; \alpha_7\rangle\} \tag{8}$$

$$C = \{\alpha_0 = \alpha_1 \to \alpha_2, \alpha_2 = \alpha_3 \to \alpha_4, \alpha_6 \to \alpha_5 \to \alpha_4 = \alpha_1, \alpha_6 = \alpha_3, \alpha_7 \to \alpha_5 = \alpha_1\}, \\ G = \{\langle x:\alpha_1, y:\alpha_3; \texttt{true}; \alpha_7\rangle\} \tag{9}$$

$$C = \{\alpha_0 = \alpha_1 \to \alpha_2, \alpha_2 = \alpha_3 \to \alpha_4, \\ \alpha_6 \to \alpha_5 \to \alpha_4 = \alpha_1, \alpha_6 = \alpha_3, \alpha_7 \to \alpha_5 = \alpha_1, \alpha_7 = \mathbb{B}\}, \\ G = \emptyset \tag{10}$$

Robinson's Algorithm:

$$\texttt{unify}(\{\alpha_0 = \alpha_1 \to \alpha_2, \alpha_2 = \alpha_3 \to \alpha_4, \alpha_6 \to \alpha_5 \to \alpha_4 = \alpha_1, \\ \alpha_6 = \alpha_3, \alpha_7 \to \alpha_5 = \alpha_1, \alpha_7 = \mathbb{B}\}) \tag{11}$$

$$\texttt{unify}(\{\alpha_2 = \alpha_3 \to \alpha_4, \alpha_6 \to \alpha_5 \to \alpha_4 = \alpha_1, \alpha_6 = \alpha_3, \alpha_7 \to \alpha_5 = \alpha_1, \alpha_7 = \mathbb{B}\})\circ \\ (\alpha_0 \mapsto \alpha_1 \to \alpha_2) \tag{12}$$

$$\texttt{unify}(\{\alpha_6 \to \alpha_6 \to \alpha_4 = \alpha_1, \alpha_6 = \alpha_3, \alpha_7 \to \alpha_5 = \alpha_1, \alpha_7 = \mathbb{B}\})\circ \\ (\alpha_2 \mapsto \alpha_3 \to \alpha_4) \circ (\alpha_0 \mapsto \alpha_1 \to \alpha_2) \tag{13}$$

$$\texttt{unify}(\{\alpha_6 = \alpha_3, \alpha_7 \to \alpha_4 = \alpha_6 \to \alpha_5 \to \alpha_5, \alpha_7 = \mathbb{B}\})\circ \\ (\alpha_1 \mapsto \alpha_6 \to \alpha_5 \to \alpha_4) \circ (\alpha_2 \mapsto \alpha_3 \to \alpha_4) \circ (\alpha_0 \mapsto \alpha_1 \to \alpha_2) \quad (14)$$

$$\texttt{unify}(\{\alpha_7 \to \alpha_5 = \alpha_3 \to \alpha_5 \to \alpha_4, \alpha_7 = \mathbb{B}\})\circ \\ (\alpha_6 \mapsto \alpha_3) \circ (\alpha_1 \mapsto \alpha_6 \to \alpha_5 \to \alpha_4) \circ (\alpha_2 \mapsto \alpha_3 \to \alpha_4) \circ (\alpha_0 \mapsto \alpha_1 \to \alpha_2) \quad (15)$$

$$\texttt{unify}(\{\alpha_7 = \mathbb{B}, \alpha_7 = \alpha_3 \to \alpha_5, \alpha_5 = \alpha_4\})\circ \\ (\alpha_6 \mapsto \alpha_3) \circ (\alpha_1 \mapsto \alpha_6 \to \alpha_5 \to \alpha_4) \circ (\alpha_2 \mapsto \alpha_3 \to \alpha_4) \circ (\alpha_0 \mapsto \alpha_1 \to \alpha_2) \quad (16)$$

$$\texttt{unify}(\{\mathbb{B} = \alpha_3 \to \alpha_5, \alpha_5 = \alpha_4\}) \circ (\alpha_7 \mapsto \mathbb{B}) \circ (\alpha_6 \mapsto \alpha_3) \\ \circ (\alpha_1 \mapsto \alpha_6 \to \alpha_5 \to \alpha_4) \circ (\alpha_2 \mapsto \alpha_3 \to \alpha_4) \circ (\alpha_0 \mapsto \alpha_1 \to \alpha_2) \quad (17)$$

ERROR: Cannot have constraint of the form $\mathbb{B} = T_1 \to T_2$! Therefore expression is not typable

3. (a) Define a successor function $\texttt{Succ} : \texttt{Nat} \to \texttt{Nat}$ that takes a number and computes its successor.

$$\texttt{Succ} = \lambda a : \texttt{Nat}.\lambda\alpha'.\lambda f : \alpha' \to \alpha'.\lambda x : \alpha'.f(a\{\alpha'\}fx)$$

And prove that it is well-typed:

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{\Gamma \vdash f : \alpha' \to \alpha'}\text{VAR} \quad \Pi_1}{\Gamma, x : \alpha' \vdash f(a\{\alpha'\}fx) : \alpha'}\text{APP}}{\Gamma, f : \alpha' \to \alpha' \vdash \lambda x : \alpha'.f(a\{\alpha'\}fx) : \alpha' \to \alpha'}\text{ABS}}{\Gamma \vdash \lambda f : \alpha' \to \alpha'.\lambda x : \alpha'.f(a\{\alpha'\}fx) : (\alpha' \to \alpha') \to \alpha' \to \alpha'}\text{ABS}}{a : \texttt{Nat} \vdash \lambda\alpha'.\lambda f : \alpha' \to \alpha'.\lambda x : \alpha'.f(a\{\alpha'\}fx) : \forall\alpha'.(\alpha' \to \alpha') \to \alpha' \to \alpha'}\text{TABS}}{\cdot \vdash \lambda a : \texttt{Nat}.\lambda\alpha'.\lambda f : \alpha' \to \alpha', \lambda x : \alpha'.f(a\{\alpha'\}fx) : \texttt{Nat} \to \texttt{Nat}}\text{ABS}$$

Where Π_1 is:

$$\dfrac{\dfrac{\dfrac{}{\Gamma \vdash f : \alpha' \to \alpha'}\text{VAR} \quad \dfrac{\Gamma \vdash a : \forall\alpha.(\alpha \to \alpha) \to \alpha \to \alpha}{\Gamma \vdash a\{\alpha'\} : (\alpha' \to \alpha') \to \alpha' \to \alpha'}\text{TAPP}}{\Gamma \vdash a\{\alpha'\}f : \alpha' \to \alpha'}\text{APP} \quad \dfrac{}{\Gamma \vdash x : \alpha'}\text{VAR}}{\Gamma \vdash (a\{\alpha'\}f)x : \alpha'}\text{APP}$$

(b) Define an addition function $\texttt{Add} : \texttt{Nat} \to \texttt{Nat} \to \texttt{Nat}$ that makes use of the successor function $\texttt{Succ}$

$$\texttt{Add} = \lambda a : \texttt{Nat}.\lambda b : \texttt{Nat}.a\{\texttt{Nat}\}\texttt{Succ}b$$

$$
\dfrac{\dfrac{\dfrac{\Pi \qquad \dfrac{\text{Proved above}}{\Gamma \vdash \texttt{Succ} : \texttt{Nat} \to \texttt{Nat}}}{\Gamma \vdash a\{\texttt{Nat}\}\texttt{Succ} : \texttt{Nat} \to \texttt{Nat}}\ \text{APP} \qquad \dfrac{}{\Gamma \vdash b : \texttt{Nat}}\ \text{VAR}}{\dfrac{a : \texttt{Nat}, b : \texttt{Nat} \vdash a\{\texttt{Nat}\}\texttt{Succ}b : \texttt{Nat}}{a : \texttt{Nat} \vdash \lambda b : \texttt{Nat}.a\{\texttt{Nat}\}\texttt{Succ}b : \texttt{Nat} \to \texttt{Nat}}\ \text{ABS}}\ \text{APP}}{\cdot \vdash \lambda a : \texttt{Nat}.\lambda b : \texttt{Nat}.a\{\texttt{Nat}\}\texttt{Succ}b : \texttt{Nat} \to \texttt{Nat} \to \texttt{Nat}}\ \text{ABS}
$$

Π :

$$
\dfrac{\dfrac{}{\Gamma \vdash a : \forall\alpha.(\alpha \to \alpha) \to \alpha \to \alpha}\ \text{VAR}}{\Gamma \vdash a\{\texttt{Nat}\} : (\texttt{Nat} \to \texttt{Nat}) \to \texttt{Nat} \to \texttt{Nat}}\ \text{TAPP}
$$