

Ej 2

a) *Reglas de tipado*

Agrego una regla de tipado para cada termino agregado a la extension

$$\frac{\Gamma \triangleright M : \sigma}{\Gamma \triangleright \text{MAT}(M) : \text{MatInf}_\sigma} (\text{T-MAT})$$

$$\frac{\Gamma \triangleright M : \text{MatInf}_\sigma \quad \Gamma \triangleright P : \sigma \quad \Gamma \triangleright N, O : \text{Nat}}{\Gamma \triangleright M[N][O] \leftarrow P : \text{MatInf}_\sigma} (\text{T-MatSet})$$

$$\frac{\Gamma \triangleright M : \text{MatInf}_\sigma \quad \Gamma \triangleright N, O : \text{Nat}}{\Gamma \triangleright M[N][O] : \sigma} (\text{T-MatGet})$$

$$\frac{\Gamma \triangleright M : \text{MatInf}_\sigma \quad \Gamma \triangleright N : \sigma \rightarrow \tau}{\Gamma \triangleright \text{map}(M, N) : \text{MatInf}_\tau} (\text{T-MatMap})$$

b) *Demostrar el siguiente juicio de tipado o explicar por que no es derivable*

$$\{i : \text{Nat}, m : \text{MatInf}_{\text{Nat}}\} \triangleright \text{map}((m[i][i] \leftarrow 0), \lambda x : \text{Nat}. \text{isZero}(x)) : \text{MatInf}_{\text{Bool}}$$

Intuitivamente deberia ser derivable, ya que mapea una matriz infinita de naturales a una matriz de booleanos que indica si cada uno era cero. Intento de demostrarlo

$$\frac{\frac{\frac{\frac{\checkmark}{m : \text{MatInf}_{\text{Nat}} \in \Gamma}}{\Gamma \triangleright m : \text{MatInf}_{\text{Nat}}} (\text{T-Var}) \quad \frac{\frac{\checkmark}{i : \text{Nat} \in \Gamma}}{\Gamma \triangleright i : \text{Nat}} (\text{T-Var}) \quad \frac{\frac{\checkmark}{\Gamma \triangleright 0 : \text{Nat}}} (\text{T-Zero})}{\Gamma \triangleright m[i][i] \leftarrow 0 : \text{MatInf}_{\text{Nat}}} (\text{T-MatSet}) \quad \frac{\frac{\frac{\checkmark}{x : \text{Nat} \in \Gamma \cup \{x : \text{Nat}\}}}{\Gamma \cup \{x : \text{Nat}\} \triangleright x : \text{Nat}} (\text{T-Var}) \quad \frac{\frac{\checkmark}{\Gamma \cup \{x : \text{Nat}\} \triangleright \text{isZero}(x) : \text{Bool}} (\text{T-IsZero})}{\Gamma \triangleright \lambda x : \text{Nat}. \text{isZero}(x) : \text{Nat} \rightarrow \text{Bool}} (\text{T-Abs})}{\Gamma = \{i : \text{Nat}, m : \text{MatInf}_{\text{Nat}}\} \triangleright \text{map}((m[i][i] \leftarrow 0), \lambda x : \text{Nat}. \text{isZero}(x)) : \text{MatInf}_{\text{Bool}}} (\text{T-MatMap})$$

c) *Indicar formalmente cómo se modifica el conjunto de valores, o explicar por qué no se modifica.*

Es necesario agregar las matrices como valor, la idea va a ser similar a como se construyen los naturales. $\text{MAT}(M)$ va a ser un valor con una matriz con todos Ms, y se va construir con $V[X][Y] \leftarrow M$. Esto quiere decir que, por ejemplo, $((\text{MAT}(0)[0][0] \leftarrow 1)[1][1] \leftarrow 2)[2][2] \leftarrow 3$ seria un valor valido. El conjunto de valores entonces sería

$$V, X, Y ::= \dots \mid \text{MAT}(M) \mid V[X][Y] \leftarrow M$$

d) *Dar la semántica operacional de a un paso para la extensión.*

$$\frac{M_1 \rightarrow M'_1}{M_1[X][Y] \leftarrow M_2 \rightarrow M'_1[X][Y] \leftarrow M_2} (\text{E-MatSetC})$$

$$\begin{array}{c}
\frac{N \rightarrow N'}{M_1[N][O] \leftarrow M_2 \rightarrow M_1[N'][O] \leftarrow M_2} \text{(E-MatSetXC)} \\
\frac{O \rightarrow O'}{M_1[N][O] \leftarrow M_2 \rightarrow M_1[N][O'] \leftarrow M_2} \text{(E-MatSetYC)} \\
\frac{}{(V[I][J] \leftarrow M)[X][Y] \rightarrow V[X][Y]} \text{(E-MatGetNoMatch)} \\
\frac{N \rightarrow N'}{M[N][O] \rightarrow M[N'][O]} \text{(E-MatGetXC)} \\
\frac{O \rightarrow O'}{M[X][O] \rightarrow M[X][O']} \text{(E-MatGetYC)} \\
\frac{M \rightarrow M'}{M[X][Y] \rightarrow M'[X][Y]} \text{(E-MatGetC)} \\
\frac{M \rightarrow M'}{(V[X][Y] \leftarrow M)[X][Y] \rightarrow (V[X][Y] \leftarrow M')[X][Y]} \text{(E-MatGetSetC)} \\
\frac{M \rightarrow M'}{(\text{MAT}(M))[X][Y] \rightarrow (\text{MAT}(M'))[X][Y]} \text{(E-MatGetMATC)} \\
\frac{}{(\text{MAT}(V))[X][Y] \rightarrow V} \text{(E-MatGetMAT)} \\
\frac{}{(V[X][Y] \leftarrow W)[X][Y] \rightarrow W} \text{(E-MatGetSet)} \\
\frac{F \rightarrow F'}{\text{map}(M, F) \rightarrow \text{map}(M, F')} \text{(E-MatMapCF)} \\
\frac{M \rightarrow M'}{\text{map}(M, F) \rightarrow \text{map}(M', F)} \text{(E-MatMapCM)} \\
\frac{}{\text{map}(\text{MAT}[M], F) \rightarrow \text{MAT}(FM)} \text{(E-MatMapMAT)} \\
\frac{}{\text{map}(V[X][Y] \leftarrow M, F) \rightarrow \text{map}(V, F)[X][Y] \leftarrow F \ M} \text{(E-MatMapSet)}
\end{array}$$

TODO: para matcheo de indices no vale poner la misma letra, hay que preguntar por igualdad sintactica con \equiv , como dice el enunciado.

e) *Mostrar como se reducen paso a paso*

$$\text{I) } (\text{MAT}(\lambda y : \text{Bool}.y)[\text{succ}(0)][0] \leftarrow \lambda x : \text{Bool}.\text{true})[0][\text{succ}(0)]$$

$$\begin{array}{l}
(\text{MAT}(\lambda y : \text{Bool}.y)[\text{succ}(0)][0] \leftarrow \lambda x : \text{Bool}.\text{true})[0][\text{succ}(0)] \\
\rightarrow_{\text{(E-MatGetNoMatch)}} (\text{MAT}(\lambda y : \text{Bool}.y))[0][\text{succ}(0)] \\
\rightarrow_{\text{(E-MatGetMAT)}} \lambda y : \text{Bool}.y
\end{array}$$

II) $((\lambda x : Bool.MAT(x)[0][0] \leftarrow false) true)[0][pred(succ(0))]$

$((\lambda x : Bool.MAT(x)[0][0] \leftarrow false) true)[0][pred(succ(0))]$
 $\rightarrow_{(E-MatGetYC), (E-PredSucc)} ((\lambda x : Bool.MAT(x)[0][0] \leftarrow false) true)[0][0]$
 $\rightarrow_{(E-MatGetC, E-AppAbs)} (MAT(\mathbf{true})[0][0] \leftarrow false)[0][0]$
 $\rightarrow_{(E-MatGetSet)} false$

III) $map(MAT(0)[0][0] \leftarrow succ(0), \lambda x : Nat.isZero(x))[0][0]$

$map(MAT(0)[0][0] \leftarrow succ(0), \lambda x : Nat.isZero(x))[0][0]$
 $\rightarrow_{(E-MatGetC, E-MatMapSet)} map(MAT(0), \lambda x : Nat.isZero(x))[0][0] \leftarrow (\lambda x : Nat.isZero(x) succ(0))[0][0]$
 $\rightarrow_{(E-MatGetC, E-MatMapMAT)} MAT((\lambda x : Nat.isZero(x)) 0)[0][0] \leftarrow (\lambda x : Nat.isZero(x) succ(0))[0][0]$
 $\rightarrow_{(E-MatGetSetC, E-AppAbs)} MAT((\lambda x : Nat.isZero(x)) 0)[0][0] \leftarrow isZero(succ(0))[0][0]$
 $\rightarrow_{(E-MatGetSetC, E-IsZeroSucc)} MAT((\lambda x : Nat.isZero(x)) 0)[0][0] \leftarrow \mathbf{false}[0][0]$
 $\rightarrow_{(E-MatGetSet, E-IsZeroSucc)} false$