

Part 1 : Theoretical Questions

שאלה 1

$$(a) \{f : [T1 \rightarrow T2], g : [T1 \rightarrow T2], a : T1\} \vdash (f (g a)) : T2$$

לא נכון.

g מקבלת פרמטר מסוג T_1 ומחזירה T_2 בניגוד לציפייה של f לקבל פרמטר מסוג T_1 .

$$(b) \{x : T1, y : T2, f : [T2 \rightarrow T1]\} \vdash (f y) : T1$$

נכון.

f מקבלת פרמטר מסוג T_2 ומחזירה T_1 שכן y מסוג T_2 .

$$(c) \{f : [T1 \rightarrow T2]\} \vdash (\text{lambda } (x) (f x)) : [T1 \rightarrow T2]$$

נכון.

f מקבלת פרמטר מסוג T_1 לכן x הוא מסוג T_1 . מאחר f מחזירה T_2 אז בהפעלת הפרוצדורה חוזר T_2 .

$$(d) \{f : [T1 * T2 \rightarrow T3]\} \vdash (\text{lambda } (x) (f x 100)) : [T1 \rightarrow T3]$$

לא נכון.

f מקבלת $T_1 * \text{Number}$ ומחזירה T_3 בניגוד לציפייה של f לקבל פרמטרים $T_1 * T_2$. כלומר ייתכן T_2 אינו Number ולכן הצהרת הטיפוסים אינה נכונה.

שאלה 2

$$(a) ((\text{lambda } (x1) (+ x1 1)) 4)$$

שלב ראשון : *Rename bound variables*:

$$((\text{lambda } (x)(+ x 1))) 4$$

שלב שני : *Assign type variables for every sub expression*:

Expression	Variable
$((\text{lambda } (x)(+ x 1))) 4$	T_0
$(\text{lambda } (x)(+ x 1))$	T_1
$(+ x 1)$	T_2
$+$	T_3
x	T_4
1	T_5
4	T_6

שלב שלישי: Construct type equations:

Sub-expressions:

Expression	Equation
$((\lambda x)(+ x 1)) 4$	$T_1 = [T_6 \rightarrow T_0]$
$(\lambda x)(+ x 1)$	$T_1 = [T_4 \rightarrow T_2]$
$(+ x 1)$	$T_3 = [T_4 * T_5 \rightarrow T_2]$

Primitives:

Expression	Equation
$+$	$T_3 = [Number * Number \rightarrow Number]$
1	$T_5 = Number$
4	$T_6 = Number$

שלב רביעי: Solve the equations:

Equation	Substitution
$T_1 = [T_6 \rightarrow T_0]$	$\{T_1 = [T_6 \rightarrow T_0]\}$
$T_1 = [T_4 \rightarrow T_2]$	
$T_3 = [T_4 * T_5 \rightarrow T_2]$	
$T_3 = [Number * Number \rightarrow Number]$	
$T_5 = Number$	
$T_6 = Number$	

Equation	Substitution
$T_3 = [T_4 * T_5 \rightarrow T_2]$	$\{T_1 = [T_6 \rightarrow T_0]\}$
$T_3 = [Number * Number \rightarrow Number]$	
$T_5 = Number$	
$T_6 = Number$	
$T_4 = T_6$	
$T_2 = T_0$	

Equation	Substitution
$T_3 = [Number * Number \rightarrow Number]$	$\{T_1 = [T_6 \rightarrow T_0], T_3 = [T_4 * T_5 \rightarrow T_2]\}$
$T_5 = Number$	
$T_6 = Number$	
$T_4 = T_6$	
$T_2 = T_0$	

Equation	Substitution
$T_5 = Number$	$\{T_1 = [T_6 \rightarrow T_0], T_3 = [T_4 * T_5 \rightarrow T_2]\}$
$T_6 = Number$	
$T_4 = T_6$	
$T_2 = T_0$	
$T_4 = Number$	
$T_2 = Number$	

<i>Equation</i>	<i>Substitution</i>
$T_4 = T_6$	$\{T_1 = [Number \rightarrow T_0], T_3 = [T_4 * Number \rightarrow T_2],$ $T_5 = Number, T_6 = Number\}$
$T_2 = T_0$	
$T_4 = Number$	
$T_2 = Number$	

<i>Equation</i>	<i>Substitution</i>
$T_2 = T_0$	$\{T_1 = [Number \rightarrow T_0],$ $T_3 = [Number * Number \rightarrow T_2],$ $T_5 = Number, T_6 = Number, T_4 = Number\}$
$T_4 = Number$	
$T_2 = Number$	

<i>Equation</i>	<i>Substitution</i>
$T_4 = Number$	$\{T_1 = [Number \rightarrow T_2],$ $T_3 = [Number * Number \rightarrow T_2],$ $T_5 = Number, T_6 = Number, T_4 = Number,$ $T_2 = T_0\}$
$T_2 = Number$	

<i>Equation</i>	<i>Substitution</i>
	$\{T_1 = [Number \rightarrow Number],$ $T_3 = [Number * Number \rightarrow Number],$ $T_5 = Number, T_6 = Number, T_4 = Number,$ $T_2 = Number, T_0 = Number\}$

$$\begin{aligned}
&\{T_1 = [Number \rightarrow Number], \\
&T_3 = [Number * Number \rightarrow Number], \\
&T_5 = Number, \\
&T_6 = Number, \\
&T_4 = Number, \\
&T_2 = Number, \\
&\mathbf{T_0 = Number}\}
\end{aligned}$$

(b) ((lambda (f1 x1) (f1 x1 1)) 4 +)

שלב ראשון: Rename bound variables :

$((\text{lambda } (f \ x) (f \ x \ 1)) \ 4 \ +)$

שלב שני: Assign type variables for every sub expression :

Expression	Variable
$((\text{lambda } (f \ x) (f \ x \ 1)) \ 4 \ +)$	T_0
$(\text{lambda } (f \ x) (f \ x \ 1))$	T_1
$(f \ x \ 1)$	T_2
f	T_3
x	T_4
1	T_5
4	T_6
$+$	T_7

שלב שלישי: Construct type equations :

Sub-expressions:

Expression	Equation
$((\text{lambda } (f \ x) (f \ x \ 1)) \ 4 \ +)$	$T_1 = [T_6 * T_7 \rightarrow T_0]$
$(\text{lambda } (f \ x) (f \ x \ 1))$	$T_1 = [T_3 * T_4 \rightarrow T_2]$
$(f \ x \ 1)$	$T_3 = [T_4 * T_5 \rightarrow T_2]$

Primitives:

Expression	Equation
$+$	$T_7 = [\text{Number} * \text{Number} \rightarrow \text{Number}]$
1	$T_5 = \text{Number}$
4	$T_6 = \text{Number}$

שלב רביעי: Solve the equations :

Equation	Substitution
$T_1 = [T_6 * T_7 \rightarrow T_0]$	$\{T_1 = [T_6 * T_7 \rightarrow T_0]\}$
$T_1 = [T_3 * T_4 \rightarrow T_2]$	
$T_3 = [T_4 * T_5 \rightarrow T_2]$	
$T_7 = [\text{Number} * \text{Number} \rightarrow \text{Number}]$	
$T_5 = \text{Number}$	
$T_6 = \text{Number}$	

<i>Equation</i>	<i>Substitution</i>
$T_3 = [T_4 * T_5 \rightarrow T_2]$	$\{T_1 = [T_6 \rightarrow T_0]\}$
$T_7 = [Number * Number \rightarrow Number]$	
$T_5 = Number$	
$T_6 = Number$	
$T_2 = T_0$	
$T_3 = T_6$	
$T_7 = T_4$	

<i>Equation</i>	<i>Substitution</i>
$T_5 = Number$	$\{T_1 = [T_6 \rightarrow T_0], T_3 = [T_4 * T_5 \rightarrow T_2],$ $T_7 = [Number * Number \rightarrow Number]\}$
$T_6 = Number$	
$T_2 = T_0$	
$T_3 = T_6$	
$T_7 = T_4$	

<i>Equation</i>	<i>Substitution</i>
$T_6 = Number$	$\{T_1 = [T_6 \rightarrow T_0],$ $T_3 = [T_4 * Number \rightarrow T_2],$ $T_7 = [Number * Number \rightarrow Number],$ $T_5 = Number\}$
$T_2 = T_0$	
$T_3 = T_6$	
$T_7 = T_4$	

<i>Equation</i>	<i>Substitution</i>
$T_2 = T_0$	$\{T_1 = [Number \rightarrow T_0],$ $T_3 = [T_4 * Number \rightarrow T_2],$ $T_7 = [Number * Number \rightarrow Number],$ $T_5 = Number, T_6 = Number\}$
$T_3 = T_6$	
$T_7 = T_4$	

<i>Equation</i>	<i>Substitution</i>
$T_3 = T_6$	$\{T_1 = [Number \rightarrow T_2],$ $T_3 = [T_4 * Number \rightarrow T_2],$ $T_7 = [Number * Number \rightarrow Number],$ $T_5 = Number,$ $T_6 = Number,$ $T_2 = T_0\}$
$T_7 = T_4$	

<i>Equation</i>	<i>Substitution</i>
$T_7 = T_4$	$\{T_1 = [Number \rightarrow T_2],$ $T_3 = [T_4 * Number \rightarrow T_2],$ $T_7 = [Number * Number \rightarrow Number],$ $T_5 = Number,$ $T_6 = Number,$ $T_2 = T_0\}$

$$[T_4 * Number \rightarrow T_2] = Number$$

הגענו לסתירה ולכן הקוד אינו תקין. (תנו לנו בונוס – תחליפו בין 4 ל+ ותקבלו קוד תקין).

Part 2.b

מוחזר $Promise<R>$ כי מאחר והמילה השמורה *async* יוצרת $Promise<T>$ היא דורשת מה *store* להחזיר $Promise$ של משהו ולכן מאחר וה-*Store* מחזיר את ה-*value* שהוא מסוג *R* אז חוזר $.Promise<R>$

Part3.1

Typing rule define:

For every: type environment $_Tenv$,
variable $_x1$
expressions $_e1$ and type expressions $_S1, _U1$:

If $_Tenv \vdash _e1 : _S1 \mid _U1$
Then $_Tenv \vdash (define _x1 _e1) : void$

Typing rule set!:

For every: type environment $_Tenv$,
variable $_x1$
expressions $_e1$
type expressions $_S1$:

If $_Tenv \vdash _x1 : _S1$
 $_Tenv \vdash _e1 : _S1$
Then $_Tenv \vdash (set! _x1 _e1) : void$