POPL

ASSIGNMENT – III

Itish Agarwal
18CS30021

Q1. (a) We have,

$\mathcal{E}_0 \cup y : bool : \; y := true$

→ From the constant rule, true : bool.

$\mathcal{E} = \mathcal{E} \cup \{true : bool\}$

→ From the assignment rule,

$$\frac{\mathcal{E} \vdash y : Ref\ bool, \quad \mathcal{E} \vdash true : bool}{\mathcal{E} \vdash y := true : command}$$

∴ Type of given expression is "command".

(b) Given

$\mathcal{E} = \mathcal{E}_0 \cup func1 : (A \rightarrow B) \cup func2 : (C \rightarrow B)$

$\lambda(x:A)\,(func\ x);\ \lambda(q:C)(func2\ q)$

→ $P = \lambda(x:A)(func1\ x)$

→ $Q = \lambda(q:C)\,(func2\ q)$

From sequencing rule,

$$\frac{\mathcal{E} \vdash M : S, \quad \mathcal{E} \vdash N : T}{\mathcal{E} \vdash M;\ N : T}$$

Scanned with CamScanner

So, we first need to find types expression for P and Q.

$\rightarrow \lambda(x:A)(\text{func1 } x)$

From function rule,

$$\frac{\mathcal{E} \cup \{x:A\} \vdash (\text{func1 } x) : T}{\mathcal{E} \vdash \lambda(x:A).\ (\text{func1 } x) : A \rightarrow T}$$

$\rightarrow$ func1 x. First we need to find this.

Application rule

$$\frac{\mathcal{E} \vdash \text{func1} : A \rightarrow B, \quad \mathcal{E} \vdash x : A}{\mathcal{E} \vdash \text{func1 } x : B}$$

$\therefore$ type of $(\text{func1 } x)$ is B

I tish Agarwal
18CS30021

$\rightarrow \lambda(x:A)(\text{func1 } x)$

From function rule

$$\frac{\mathcal{E} \cup \{x:A\} \vdash (\text{func1 } x) : B}{\mathcal{E} \vdash \lambda(x:A).(\text{func1 } x) : A \rightarrow B}$$

$\therefore$ type of $(\lambda(x:A)(\text{func1 } x))$ is $A \rightarrow B$

$\rightarrow \lambda(q:C)(\text{func2 } q)$

From function rule, $\mathcal{E} = \mathcal{E} \cup \{q:C\}$

Scanned with CamScanner

→ Application rule for (func2 q)

$$\frac{\Sigma \vdash func2 : C \to B \;,\; \Sigma \vdash q : C}{\Sigma \vdash func2\ q : B}$$

Itish Agarwal
18CS30021

$\therefore\ \Sigma \cup \{func2q : B\}$

→ $\lambda(q : C)\,(func2\ q)$

Function rule :

$$\frac{\Sigma \cup \{q : C\} \vdash (func2\ q) : B}{\Sigma \vdash \lambda(q : C)\,(func2\,q) : (C \to B)}$$

$\lambda(x : A)\,(func1\ x)\ ;\ \lambda(q : C)\ \cancel{z}.(func2\ q)$

From sequencing rule,

~~Otototo~~

$$\frac{\Sigma \vdash (\lambda(x : A)(func1\ x)) : A \to B \;,\; \Sigma \vdash (\lambda(q : C)(func2\ q)) : (C \to}{\Sigma \vdash (\lambda(x.A)(func1\ x)\ ;\ \lambda(q : C)(func2\ q)) : C \to B}$$

$\therefore$ Hence type of given expression is

$C \to B$

Scanned with CamScanner

(c) Given:

$\mathcal{E} = \mathcal{E}_0 \cup \{ | : Bool \rightarrow Bool \rightarrow Bool \} \cup \{ true : Bool \}$

$\rightarrow \lambda(\omega : Bool \rightarrow \pi).(\lambda(x : Bool).(\omega(x | true)))$

By function rule,

$\mathcal{E} = \mathcal{E} \cup \{ \omega : Bool \rightarrow \pi \}$ where,

π is given by:

By π constant rule

$\mathcal{E} = \mathcal{E} \cup \{ \pi : float \}$

$\therefore \mathcal{E} = \mathcal{E} \cup \{ \omega : Bool \rightarrow float \}$

$\rightarrow \lambda(x : Bool).(\omega(x | true))$

By function rule

$\mathcal{E} = \mathcal{E} \cup \{ x : Bool \}$

$\rightarrow (x | true)$

By application rule,

$\mathcal{E} = \mathcal{E} \cup \{ (x | true) : Bool \}$

$\rightarrow (\omega(x | true))$

By application rule,

Itish Agarwal
18CS30021

Scanned with CamScanner

$$\mathcal{E} = \mathcal{E} \cup \{ w\ (x \mid true) : float \}$$

$\rightarrow \lambda\ (x : Bool).\ (w\ (x \mid true))$

By function rule,

$$\mathcal{E} = \mathcal{E} \cup \{ \lambda (x : Bool).(w\ (x \mid true)) : Bool \rightarrow float \}$$

$\rightarrow \lambda (w : Bool \rightarrow \pi).\ \lambda (x : Bool).\ (w\ (x \mid true))$

By function rule,

$$\mathcal{E} = \mathcal{E} \cup \{ \lambda (w : Bool \rightarrow \pi). \lambda (x : Bool).\ (w (x \mid true)) : (Bool \rightarrow Float) \rightarrow (Bool \rightarrow Float) \}$$

Hence type of given expression is:

$(Bool \rightarrow Float) \rightarrow (Bool \rightarrow Float)$

Itish Agarwal
18CS30021

(d) Given

$$\mathcal{E} = \mathcal{E}_0 \cup \{ + : S \rightarrow S \}$$

$\rightarrow \lambda\ (f : S \rightarrow C).\ \lambda (x : S).\ f\ (+\ x)$

From function rule,

$$\mathcal{E} = \mathcal{E} \cup \{ f : S \rightarrow C \}$$

Scanned with CamScanner

$\rightarrow \lambda(x:S).f(+x)$

From function rule,

$\mathcal{E} = \mathcal{E} \cup \{(+x) : S\}$

$\rightarrow f(+x)$

From application rule,

Itish Agarwal
18CS30021

$\mathcal{E} = \mathcal{E} \cup \{(+x) : S\}$

$\rightarrow f(+x)$

From application rule,

$\mathcal{E} = \mathcal{E} \cup \{f(+x) : C\}$

$\rightarrow \lambda(x:S).f(+x)$

From function rule,

$\mathcal{E} = \mathcal{E} \cup \{\lambda(x:S).f(+x) : S \rightarrow C\}$

$\rightarrow \lambda(f:S\rightarrow C).\lambda(x:S).f(+x)$

from function rule,

$\mathcal{E} = \mathcal{E} \cup \{\lambda(f:S\rightarrow C).\lambda(x:S).f(+x) : (S\rightarrow C)\rightarrow(S\rightarrow C)\}$

Hence type of given expression is:

$(S\rightarrow C) \rightarrow (S\rightarrow C)$

Scanned with CamScanner

(e) Given

$\mathcal{E}_0 = \{x : \text{Ref Bool},\ y : \text{Bool}\}$

$\mathcal{E} = \mathcal{E}_0 \cup \{succ : Int \to Int\} \cup \{true : Bool\} \cup \{4 : Int\}$

$\rightarrow$ succ 4

From application rule

$\mathcal{E} = \mathcal{E} \cup \{succ\ 4 : Int\}$

Itish Agarwal
18CS30021

$\rightarrow$ $x := true$

from assignment rule,

$\mathcal{E} = \mathcal{E} \cup \{x := true : command\}$

$\rightarrow$ succ 4; $x := true$

From sequencing rule,

$\mathcal{E} = \mathcal{E} \cup \{succ\ 4;\ x := true : Command\}$

Hence type of given expression is

= "Command".

Scanned with CamScanner

Q2.

(a) Given

$$\mathcal{E} = \mathcal{E}_0 \cup \{ \phi : Float \to Integer \}$$

$\to \lambda (p : Float \to Integer). \lambda (f : Float \to Float). \lambda (y : Float). \; p(f(f(y)))$

From function rule,

$$\mathcal{E} = \mathcal{E} \cup \{ p : Float \to Integer \}$$

$\to \lambda (f : Float \to Float). \lambda (y : Float). \; p \; f(f(y))$

$\mathcal{E} = \mathcal{E} \cup \{ f : Float \to Float \}$ (By function rule)

$\to \lambda (y : Float). \; p \; f(f(y))$

By function rule,

Itish Agarwal
18CS30021

$$\mathcal{E} = \mathcal{E} \cup \{ y : Float \}$$

$\to (f \; y)$

By application rule,

$$\cancel{\mathcal{E} = \mathcal{E} \cup \{ p(f(f(y))) : Integer \}}$$

$$\mathcal{E} = \mathcal{E} \cup \{ fy : Float \}$$

Scanned with CamScanner

$\rightarrow f\,(f\,y)$

By application rule,

$\mathcal{E} = \mathcal{E} \cup \{ f(fy) : Float \}$

Itish Agarwal
18CS30021

$\rightarrow p\,(f\,(f\,y))$

By application rule,

$\mathcal{E} = \mathcal{E} \cup \{ p\,(f\,(f\,y)) : Integer \}$

$\rightarrow \lambda\,(y : Float).\, p(f\,(f\,y))$

By function rule

$\mathcal{E} = \mathcal{E} \cup \{ \lambda\,(y : Float).\, p(f\,(fy)) : Float \rightarrow Integer \}$

$\rightarrow \lambda(f : Float \rightarrow Float).\, \lambda\,(y : Float).\, p\,(f\,(f\,y))$

By function rule,

$\mathcal{E} = \mathcal{E} \cup \{ \lambda(f : Float \rightarrow Float).\, \lambda(y : Float).\, p\,(f\,(fy))$
$: (Float \rightarrow Float) \rightarrow (Float \rightarrow Integer) \}$

Scanned with CamScanner

$\to \lambda(p: Float \to Integer). \lambda(f: Float \to Float). \lambda(y: Float). p(f(f(y)))$

By function rule,

$\mathcal{E} = \mathcal{E} \cup \{\lambda(p: Float \to Integer). \lambda(f: Float \to Float). \lambda(y: Float)$

$\quad . p\, f(fy)) : (Float \to Integer) \to ((Float \to Float)$

$\quad\quad \to (Float \to Integer))\}$

$\to (\lambda(p: Float \to Integer) \ldots\ldots\ldots p(f(fy))) \phi$

By application rule,

$\mathcal{E} = \mathcal{E} \cup \{(\lambda(p: Float \to Integer) \ldots\ldots p\, f(fy)) \phi :$

$\quad (Float \to Float) \to (Float \to Integer)\}$

Hence, type of given expression is:

$(Float \to Float) \to (Float \to Integer)$

(b)

PTO

Itish Agarwal
18CS30021

Scanned with CamScanner

(b) Given

$$\Sigma = \Sigma \cup \{\phi : Bool \to Bool \to Bool\} \cup \{true : Bool\}$$

$$\to \lambda\,(func1 : Bool \to Char).\lambda\,(\Gamma : Bool).\,func1\,(\Gamma\,\phi\,true)$$

By function rule,

$$\Sigma = \Sigma \cup \{func1 : Bool \to Char\}$$

$$\to \lambda\,(\Gamma : Bool).\,func1\,(\Gamma\ \phi\ true)$$

By function rule,

$$\Sigma = \Sigma \cup \{\Gamma : Bool\}$$

Itish Agarwal
18CS30021

$$\to \Gamma\ \phi\ true$$

By application rule,

$$\Sigma = \Sigma \cup \{\Gamma\,\phi\,true : Bool\}$$

$$\to func1\ (\Gamma\ \phi\ true)$$

By application rule,

$$\Sigma = \Sigma \cup \{func1\ (\Gamma\,\phi\,true) : Char\}$$

Scanned with CamScanner

$\rightarrow \lambda (r : Bool).\ func1\ (\cancel{\#} r\ \phi\ true)$

By function rule,

$\mathcal{E} = \mathcal{E} \cup \{\lambda (r : Bool).\ func1\ (\cancel{\#} r\ \phi\ true) : Bool \rightarrow Char\}$

$\rightarrow \lambda (func1 : Bool \rightarrow Char).\ \lambda (r : Bool).\ func1 (r\ \phi\ true)$

By function rule,

$\mathcal{E} = \mathcal{E} \cup \{\lambda (func1 \ldots\ldots\ldots\ldots (r\ \phi\ true) : (Bool \rightarrow Char)$
$\rightarrow (Bool \rightarrow Char)\}$

Hence type of given expression is:

$(Bool \rightarrow Char) \rightarrow (Bool \rightarrow Char)$

Itish Agarwal
~~18CS~~
18CS30021

Scanned with CamScanner