

# 第 13 回課題

09430509 今田将也

## 3.5.4 3 fun A x y z = z y x

関数Aをt1として, x:t2, y:t3, z:t4, z y x:t5とすると,

t1 = t2 -> t3 -> t4 -> t5となり, t4 = t3 -> t2 -> t5となるため

したがって,

```
t1 = t2 -> t3 -> (t3 -> t2 -> t5) -> t5 = 'a -> 'b -> ('b -> 'a -> 'c) -> 'c
```

## 3.5.4 4 fun B f g = f g g

関数Bをt1として, f:t2, g:t3, f g g:t4とすると,

t1 = t2 -> t3 -> t4となり, t2 = t3 -> t3 -> t4となるため

したがって,

```
t1 = (t3 -> t3 -> t4) -> t3 -> t4 = ('a -> 'a -> 'b) -> 'a -> 'b
```

## 3.5.7

### 2 fun f x y z = x (y z) : int

f:t1, x:t2, y:t3, z:t4, (y z):t5, x (y z):t6と仮定すると. t1 = t2 -> t3 -> t4 -> t6であり, また, t3 = t4 -> t5であり, そしてt2 = t5 -> t6から,

```
t1 = (t5 -> t6) -> (t4 -> t5) -> t4 -> t6 = ('a -> 'b) -> ('c -> 'a) -> 'c -> 'b
```

そしてintはt6に適用されるので,

```
t1 = ('a -> int) -> ('c -> 'a) -> 'c -> int = ('a -> int) -> ('b -> 'a) -> 'b -> int
```

### 3 fun f x y z = (x y z) : int

f:t1, x:t2, y:t3, z:t4, (x y z):t5と仮定すると. t1 = t2 -> t3 -> t4 -> t5であり, また, t2 = t3 -> t4 -> t5であるから,

```
t1 = (t3 -> t4 -> t5) -> t3 -> t4 -> t5 = ('a -> 'b -> 'c) -> 'a -> 'b -> 'c
```

そしてt5に, intを指定するので,

```
t1 = ('a -> 'b -> int) -> 'a -> 'b -> int
```

4 fun f x y z = x y (z: int)

これは3の関数から, この関数ではt4にintを指定するので,

```
t1 = ('a -> int -> 'b) -> 'a -> int -> 'b
```

5 fun x y z = x (y z : int)

これは2の関数から, この関数ではt5にintを指定するので,

```
t1 = (int -> 'a) -> ('b -> int) -> 'b -> 'a
```

## 問3.11

1 fn x => x > 1

xは1というint型との比較に利用しているので, int型で, 結果はboolとして返るはず. したがって,

```
val it = fn : int -> bool
```

2 fn x => fn y => fn z => (x y, x "Ada", y > z)

これは, fn x y z = (x y, x "Ada", y > z)と, とれる.

式全体をt1として, x:t2,y:t3,z:t4,x y:t5,x "Ada":t6,y > z:t7,(x y, x "Ada", y > z):t8と仮定する.

t1 = t2 -> t3 -> t4 -> t8である. そして, t2 = t3 -> t5, t2 = String -> t6, t7 = boolである. また, t8は引数が3つあるので, t8 = t5 \* t6 \* t7. また, t5 = t6, t3 = Stringとすると,

```
t1 = (string -> t6) -> string -> string -> (t6 * t6 * bool) = (string -> 'a) ->
string -> string -> ('a * 'a * bool)
```

## 問題5.6.4

```
fun add1 x = x + 1; val add1 = fn : int -> int
```

```
fun comp F G =
  let
    fun C x = G(F(x))
  in
    C
  end
;
val comp = fn : ('a -> 'b) -> ('b -> 'c) -> 'a -> 'c
```

a) val compA1 = comp add1;

```
(int -> 'a) -> int -> 'a
```

関数compA1は引数として関数F (x) を取り,G (x) = F (x + 1) となるような関数G (x) ここで, xはintでなければならず, 言い換えると, compA1は関数F (x) をF (x + 1) に変換する.

b) val compCompA1 = comp compA1;

```
((int -> 'a) -> 'b) -> (int -> 'a) -> 'b
```

c) val f = compA1 add1;

```
val f = fn : int -> int
```

d) f(2);

```
val it = 4 : int
```

e) val g = compCompA1 compA1;

```
(int -> 'a) -> int -> 'a
```

関数gは関数F(x+2)をとる.

f) val h = g add1;

```
int -> int
```

関数hは整数xを取り、 $x+3$ を生成します。

g) `h(2);`

```
val it = 5 : int
```