Date: 12th September 2003 | QUIZ 2 | Total Marks: 45

1. Does the following recursive equation define for `m` a function of the type `Integer -> Integer -> Integer`? Carefully explain your answer. (15 Marks)

$$\begin{aligned} m\ \ x\ \ y\ &|\ x == y = y+1 \\ &|\ otherwise = m\ \ x\ \ (m\ \ (x-1)\ \ (y+1)) \end{aligned}$$

Soln: No it does not define a function, it defines a relation (or more than one functions, if you wish to think of it that way).

As a first attempt to define a function that satisfies the specification above, we think of one which is independent of the second argument. Assuming that this property holds for the inner calls, the specification rewrites to:

$$\begin{aligned} m\ \ x\ \ y\ &|\ x == y = y+1 \\ &|\ x \neq y = g\ \ x\ \text{, for some } g \end{aligned}$$

But the outer call is still not independent of y, To do this rewrite $x+1$ to $y+1$ and make an assumption that the inner call returns $x+1$, ie, replace $g\ \ x$ by $x+1$

$$\begin{aligned} m\ \ x\ \ y\ &|\ x == y = x+1 \\ &|\ x \neq y = x+1 \end{aligned}$$

i.e.

$$m\ \ x\ \ y = x+1$$

This is the first function that satisfies the specifications.

To find a second function which satisfies the specs, notice that the minimal set of (x, y) for which $m\ x\ y$ is constrained to be $x+1$ is $\{(x,y) \mid x \geq y \wedge even(x-y)\}$. For the rest of (x, y)s let us a give values which are different from $x+1$.

A bit of thinking will suggest that a good value to try is $m\ \ x\ \ y = y-1$. Then for $x < y$

$$\begin{aligned} & m\ \ x\ \ y \\ = & m\ \ x\ \ (m\ \ (x-1)\ \ (y+1)) \\ = & m\ \ x\ \ y \text{ -- because } (x-1) < (y+1) \\ = & y-1 \end{aligned}$$

What we have done is to verify is that assuming the constraint $x < y$ is satisfied for the main call, the inner calls too satisfy $x < y$, and assuming that the inner calls return the second argument decremented by 1, the main call also returns $y-1$.

However we should also verify the same for x and y such that $odd(x-y)$.

$$\begin{aligned} & m\ \ x\ \ y \\ = & m\ \ x\ \ (m\ \ (x-1)\ \ (y+1)) \\ = & m\ \ x\ \ y \text{ -- because } odd(x-y) \Rightarrow odd((x-1)-(y+1)) \\ = & x+1 \end{aligned}$$

Thus a second function which satisfies the specification is:

$$\begin{aligned} m\ \ x\ \ y\ \mid\ & x \geq y \wedge even(x-y) = x+1 \\ \mid\ & otherwise = y-1\ , \end{aligned}$$

2. Assuming the usual Church representation of numerals, define the squaring function by substituting suitable lambda terms for f, x and y: (15 Marks)

$$square\ \ n = f\ (n\ \ x\ \ y)$$

Soln: $square\ \ n = snd\ \ (n\ \ \lambda(x,y).(x+1,\ \ x+1*x+1)\ \ (0,0))$

Other solutions are possible.

3. Given the following two definitions in Haskell

```
data Bintree a = Node a (Bintree a) (Bintree a) | Leaf a
bot = bot
```

What is the value of each of the let expressions given below: (15 Marks)

(a)
```
let f ~(Node 1 t1 t2) = t1
in f (Node 2 (Leaf 1)(Leaf 2))
```
Ans: $\perp$

(b)
```
let f ~(Node ~1 t1 t2) = t1
in f (Node 2 (Leaf 1)(Leaf 2))
```
Ans: $Leaf\ \ 1$

(c)
```
let f ~(Node ~1 t1 t2) = t1
in f (Node 2 (Leaf bot)(Leaf 2))
```
Ans: $Leaf\ \ \perp$

(d)
```
let f ~(Node ~1 t1 t2) = g t1
g (Leaf x) = 0
in f (Node 2 (Leaf bot)(Leaf 2))
```
Ans: 0

(e)
```
let f ~(Node 1 t1 t2) = g t1
g (Leaf x) = 0
in f (Node 2 (Leaf bot)(Leaf 2))
```
Ans: $\perp$

(f)
```
let f ~(Node 1 t1 t2) = g t1
g x = 0
in f (Node 2 (Leaf bot)(Leaf 2))
```
Ans: 0

(g)
```
let f ~(Node ~1 t1 t2) = t2
in  f (Node 2 (Leaf bot)(Leaf 2))
```
Ans: $Leaf\ \ 2$