POPL Quiz 2

---

1. Assuming initial type environment

$$\Gamma = \left\{ \begin{array}{rcl} (\circ) & :: & \forall\beta\gamma\alpha \cdot (\beta \to \gamma) \to (\alpha \to \beta) \to \alpha \to \gamma, \\ \text{map} & :: & \forall\alpha\beta \cdot (\alpha \to \beta) \to [\alpha] \to [\beta] \end{array} \right\}$$

show the steps (preferably a box diagram) to infer the type for `map2` for the following program:

```
let
   map2 = map ∘ map
in
   map2
```

Recall that $\circ$ is function composition. `let` is the non-recursive let. (15 Marks)

2. *Error handling using Monads:* Consider the following data constructor representing the expressions involving composition of log and sqrt functions.

```
data LogSqrt = Val Float | Log LogSqrt | Sqrt LogSqrt
```

Use monads to define an evaluation function `eval` for `LogSqrt` in which failures/errors are handled using `Maybe` monad. Assume the existence of library finctions `log` and `sqrt` that operate on floating point values. (15 Marks)

The end