

Exam

Mark each answer with your initials. Write clearly and comment what you do, that might give you points even if the result is wrong.. Each question is worth five points.

1. Rewrite the following function with a list comprehension and state its type:

```
g x = map ($x)
```

2. State the relationship between the two functions `zip` and `unzip`?

3. Define the following function using pattern matching:

```
oneOf :: Bool -> Bool -> Bool -> Bool
oneOf a b c
  | not (a or b) = c
  | not (b or c) = a
  | not (a or c) = b
  | otherwise   = False
```

4. What is the type and the value of the expression:

```
do [1,2,3]; "lambda"
```

5. Explain the class concept in Haskell. If the language did not have it, what would the consequences be?

6. Explain what the following function does

```
c a = (a\\) . (a\\)
```

The function `(\\)` is defined in Haskell's List library as

```
(\\) = foldl (flip delete)
```

and

```
delete x []      = []
delete x (y:ys)  = if x == y then ys else y : delete x ys
```