

Exam

Answers may be given in either Swedish or English.

1. Rewrite the following definition so that only the function name appears to the left hand side of the equation and that the right hand side is not a lambda expression:

```
f x = (5 - x) / 3
```

Do the same thing for the following definition:

```
f x y = (5 - x) / y
```

2. Given the following function:

```
f x y = do  
  a <- x  
  b <- y  
  return a+b
```

(a * b)

What is the type of `f`?

What is the value of `f [1,2,3] [2,4,8]`?

What is the value of `f (Just 5) Nothing`?

3. Haskell defines functions in so called *curried form*. Explain what that means and what its main advantage is.

4. The standard prelude contains the function

```
replicate :: Int -> a -> [a]
```

The following might seem like reasonable definition for it

```
replicate n x = take n [x,x,..]
```

But it is actually not sufficient. Why not?

5. Give the types of these three expressions. Also explain what each of the expressions mean.

```
zipWith map  
map zipWith  
map.zipWith
```

6. The standard prelude contains two similar functions *foldl* and *foldr*.

Explain how they differ?

Give their respective types.

One of them can be used to implement the standard function *map*.

Give that definition, i.e. it should be on the form:

```
map f = foldr ... or map f = foldl ...
```