

# 1 COMP304 Week 4 Exercises

## Previous Week

### 1.0.1 Ex 2

Convert the following function to not use guard statements (Hint: Use pattern matching)

```
fibb :: Int -> Int
fibb a | a == 0 = 0
      | a == 1 = 1
      | otherwise = fibb (a-1) + fib (a-2)
```

### 1.0.2 Ex 3

Write a simple guess my number game answer checking function. Give it using both pattern matching and guards.

The number guessing should be hardcoded, and the same for each attempt.

You should return “You have found my number” on a correct guess

On an error you should return “That’s not my number”.

For an extension you should return whether or not the fixed number is higher or lower than the guess.

```
guess :: Int -> String
```

### 1.0.3 Ex 6

Write a function to reverse the order of a list

```
reverse :: [a] -> [a]
```

## 2 This Week

### 2.1 Ex 1

Write a function that returns the nth element in a list

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

### 2.2 Ex 2 - Hard

Write a function to return the maximum value of a list using recursion. Hint: you might need to keep things in the list

```
max :: [Int] -> Int
```

### 2.3 Ex 3

Write a function to return the maximum value of a list using higher order functions, using a fold method is highly recommended

```
maxF0 :: [Int] -> Int
```

### 2.4 Ex 4 - Very Hard

Write a function which returns the **greatest difference** between the value of two consecutive elements in a list

```
maxDiff :: [Int] -> Int
```

Hint: This will use a lot of more complex higher order functions, using things such as

```
zip :: [a] -> [b] -> [(a,b)]
```

and

```
foldl :: (b -> a -> b) -> b -> [a] -> b
```