

Übung: Functors

- Die Functor Typklasse

1 Aufgabe

Gegeben ist folgender Datentyp und Functor Instanz:

```
1 newtype Boxed a = Boxed {unbox :: a}
2
3 instance Functor Boxed where
4     fmap f (Boxed a) = Boxed (f a)
```

Beweisen Sie im Detail (die “Functor Laws”):

```
1     fmap id b = b
2     fmap (f . g) b = fmap f $ fmap g b
```

respektive:

```
1     id <$> b = b
2     (f . g) <$> b = f <$> (g <$> b)
```

2 Aufgabe

Gegeben ist der Datentyp

```
1 newtype FromInt b = FromInt {fun :: Int -> b}
```

- Schreiben Sie eine Functor Instanz für FromInt.
- Beweisen Sie die Functor Regeln.

3 Aufgabe

Die Klasse `Contravariant` ist fast gleich wie die Klasse `Functor` definiert, nur etwas “verdreht”. Anstelle der Funktion

```
1     fmap :: (a -> b) -> f a -> f b
```

muss man eine Funktion

```
1     contramap :: (b -> a) -> f a -> f b
```

implementieren. Definieren Sie einen Datentyp und eine dazu passende Instanz für `Contravariant`.