Dr. Th. Letschert
Softwarekomponenten

THM TECHNISCHE HOCHSCHULE MITTELHESSEN
MNI Mathematik, Naturwissenschaften und Informatik

Aufgabenblatt 11

Aufgabe 1

Ein grauhaariger, zottelbärtiger Java-Veteran, der vor 25 Jahren von C umgeschult wurde, hat folgendes **imperative** 100% handgestricktes nahezu naturreines[1] Programm abgeliefert:

```
def isPrimeAsync(n: Long): Future[Boolean] = {
  val promise = Promise[Boolean]
  val thread = new Thread(
    new Runnable {
      override def run(): Unit = {
        var count = 0
        for (i <- Range.Long(2L, n / 2 + 1, 1)) {
          if (n % i == 0) count = count+1
        }
        promise.success(count == 0)
      }
    }
  )
  thread.start()
  promise.future
}

def toLong(str: String): Option[Long] =
  try {
    Some(str.toLong)
  } catch {
    case e: NumberFormatException => None
  }

def countFactorsAsync(str: String): Future[Option[String]] = {
  val promise = Promise[Option[String]]
  val xO = toLong(str)
  if (xO.isDefined) {
    val x = xO.get
    val pF: Future[Boolean] = isPrimeAsync(x)
    val thread = new Thread(
      new Runnable {
        override def run(): Unit = {
          val b = Await.result(pF, 1000.second)
          val bStr = if (b) s"$str is prime" else s"$str is not prime"
          promise.success(Some(bStr))
        }
      }
```

[1] *Promise* und *Future* sind die einzige synthetischen Fasern. Wer Spaß an den *Basics* hat, kann sie auch durch selbst gesponnene naturreine Java-Wolle ersetzen.

```
    )
    thread.start()
  } else {
    promise.success(None)
  }
  promise.future
}

val res = // 10000079 is prime
  countFactorsAsync("10000079").onComplete {
    case Success(value) =>
      value match {
        case Some(str) => println(str)
        case None => println("Some Error occured")
      }
    case Failure(e) => println(e)
  }

def main(args: Array[String]): Unit = {
  Thread.sleep(3000)
  println(res)
}
```

Machen Sie etwas zeitgemäßes und **funktionals** daraus – Monaden und Monadentransformer sollten auch darin vorkommen.

Worin besteht Ihrer Meinung nach der Fortschritt in der Softwaretechnik, wenn es überhaupt einen gibt?

Aufgabe 2

Definieren Sie den Monadentransformer *EitherT* für *Either* und geben Sie ein Verwendungsbeispiel für Ihren Transformer an.

Aufgabe 3

Lesen Sie *ListT done right*[2], eine Diskussion zum Monadentransformer *ListT* in Haskell. Definieren Sie dann – von der Lektüre inspiriert – einen Monadentransformer *ListT* für *List*.

[2] https://wiki.haskell.org/index.php?title=ListT_done_right