

7.1 Lazy Evaluation

a) Was beschreibt in der funktionalen Programmierung Lazy-Evaluation? Geben Sie ein Scala-Beispiel einer lazy-Funktion an, welches Parameter als call-by-name übergibt.

Lazy-Evaluation verschiebt die Evaluation eines Wertes auf den Zeitpunkt an dem man den Inhalt tatsächlich das erste mal braucht und nicht im Moment der Deklaration.

```
// In diesem Fall wird die Methode x zweimal aufgerufen.  
def lazyFoo(x: => Int, y: Int) = (x * x) / y  
  
def retX(): Int {  
  print(100)  
  100  
}  
  
lazyFoo(retX(), 200)
```

b) Was ist bezogen auf dieses Beispiel der Vorteil gegenüber call-by-value oder call-by-reference?

Die via Name übergebene Methode wird bei jedem Vorkommen aufgerufen. Dadurch können z.B. Vorbedingungen jedesmal geprüft werden. Seiteneffekte treten jedoch auch pro Aufruf auf.