

Programmierung 2 - Sommersemester 2019

Prof. Dr. Markus Esch

Übungsblatt Nr. 17 Abgabe KW 20

1. Aufgabe

- (a) Implementieren Sie eine Methode `applyAndPrint`, die zwei natürliche Zahlen i und j sowie eine Funktion $f : \mathbb{N} \rightarrow \mathbb{N}$ entgegen nimmt. Die Methode soll die übergebene Funktion auf alle Zahlen zwischen i und j anwenden und das Ergebnis ausgeben. Nutzen Sie zur Übergabe der Funktion folgendes funktionales Interface:

```
public interface MyFunction{  
    public int apply(int i);  
}
```

- (b) Rufen Sie die `applyAndPrint`-Methode mit den im Folgenden aufgeführten Funktionen auf. Implementieren Sie die Funktionen jeweils als Lambda-Ausdruck und als anonyme Klasse. Implementieren Sie die Funktion ii zusätzlich als Top-Level-Klasse und als Static Nested Class. Vergleichen Sie den Implementierungsaufwand.

i $f(x) = x^2$

ii $f(x) = x!$

iii $f(x) = x^{(x+1)}$

iv $f(x) = \text{fib}(x)$ (Fibonacci-Folge)

- (c) Implementieren Sie ein funktionales Interface, welches das in Aufgabe 1a verwendete Interface erweitert. Dieses Interface soll die beiden folgenden zusätzlichen Methoden mit Default-Implementierung besitzen:

i `conditionateInput`: Diese Methode nimmt ein Prädikat als Argument entgegen und gibt einen Lambda-Ausdruck zurück. Der Lambda-Ausdruck prüft für einen gegebenen Integer-Wert, ob dieser das Prädikat erfüllt. Ist das Prädikat erfüllt, wird das Ergebnis der `apply`-Methode des Interfaces zurückgegeben, ansonsten wird 0 zurückgegeben.

ii `conditionateOutput`: Diese Methode nimmt ebenfalls ein Prädikat als Argument entgegen und gibt einen Lambda-Ausdruck zurück. Der Lambda-Ausdruck prüft, ob das Ergebnis der `apply`-Methode das Prädikat erfüllt. Ist das Prädikat erfüllt, wird das Ergebnis zurückgegeben, ansonsten 0.

- (d) Implementieren Sie ein Prädikat `odd` und ein Prädikat `even`. Die Prädikate sollen prüfen, ob ein gegebener Integer-Wert gerade bzw. ungerade ist. Implementieren Sie das `odd`-Prädikat als anonyme Klasse und das `even`-Prädikat als Lambda-Ausdruck.
- (e) Rufen Sie die Methode `applyAndPrint` mit einem Lambda-Ausdruck auf, welcher für gerade Zahlen die Quadratzahl berechnet.

- (f) Rufen Sie die Methode `applyAndPrint` mit einem Lambda-Ausdruck auf, welcher prüft, ob die Fakultät einer gegebenen Zahl ungerade ist. Ist dies der Fall, wird die Fakultät zurückgegeben, ansonsten 0.