

Informatik I: Einführung in die Programmierung

Prof. Dr. Peter Thiemann
Dr. Daniel Büscher, Hannes Saffrich
Wintersemester 2019

Universität Freiburg
Institut für Informatik

Übungsblatt 3 – Lösungen

Abgabe: Montag, 11.11.2019, 9:00 Uhr morgens

Legen Sie in Ihrer Arbeitskopie ein Verzeichnis `sheet03` an und bearbeiten Sie dort die folgenden Aufgaben. Vergessen Sie nicht das Verzeichnis und die enthaltenen Dateien zur Arbeitskopie hinzuzufügen und mit `svn commit` zu publizieren, sonst können Ihre Abgaben nicht bearbeitet werden.

Achten Sie darauf Dateien, Verzeichnisse und Funktionen exakt wie in den Aufgaben zu benennen. **Aufgaben mit falschen Namen werden ab diesem Blatt mit 0 Punkten bewertet**, da diese der Teilautomatisierung der Korrektur im Weg stehen.

Aufgabe 3.1 (Fahrenheit Reloaded; Datei: `fahrenheit.py`; Punkte: 3+3+1+3)

- (a) Schreiben Sie eine Funktion `fahrenheit_to_celsius`, die einen Fahrenheit-Wert (`float`) als Argument nimmt und den entsprechenden Celsius-Wert (`float`) ungerundet zurückgibt.
- (b) Schreiben Sie eine Funktion `celsius_to_kelvin`, die einen Celsius-Wert (`float`) als Argument nimmt und den entsprechenden Kelvin-Wert (`float`) ungerundet zurückgibt.
- (c) Schreiben Sie eine Funktion `fahrenheit_to_kelvin`, die einen Fahrenheit-Wert (`float`) als Argument nimmt und den entsprechenden Kelvin-Wert (`float`) ungerundet zurückgibt. Verwenden Sie hierzu die Funktionen `fahrenheit_to_celsius` und `celsius_to_kelvin`, anstatt die Konvertierung erneut zu implementieren.
- (d) Verwenden Sie `input`, um den Benutzer nach einem Fahrenheit-Wert zu fragen. Verwenden Sie `fahrenheit_to_celsius` und `fahrenheit_to_kelvin`, um die entsprechenden Celsius- und Kelvin-Werte zu berechnen. Verwenden Sie `print` um die Celsius- und Kelvin-Werte, auf zwei Nachkommastellen gerundet, dem Benutzer zu präsentieren.

Ein Aufruf des Scripts könnte wie folgt aussehen (Benutzereingabe in grün):

```
$ python3 fahrenheit.py
Fahrenheit: 9.11
Celsius: -12.72
Kelvin: 260.43
```

Hinweis: Die Datei `fahrenheit.py` sollte also folgende Form haben:

```
def fahrenheit_to_celsius(fahrenheit):  
    [...]
```

```
def celsius_to_kelvin(celsius):  
    [...]
```

```
def fahrenheit_to_kelvin(fahrenheit):  
    [...]
```

```
fahrenheit = input([...])  
[...]  
print([...])
```

Lösung:

```
def fahrenheit_to_celsius(fahrenheit):  
    return (fahrenheit - 32) * (5 / 9)
```

```
def celsius_to_kelvin(celsius):  
    return celsius + 273.15
```

```
def fahrenheit_to_kelvin(fahrenheit):  
    return celsius_to_kelvin(fahrenheit_to_celsius(fahrenheit))
```

```
fahrenheit = float(input("Fahrenheit: "))
```

```
celsius = fahrenheit_to_celsius(fahrenheit)  
kelvin = fahrenheit_to_kelvin(fahrenheit)
```

```
print("Celsius:", round(celsius, 2))  
print("Kelvin:", round(kelvin, 2))
```

Aufgabe 3.2 (Osterformel; Datei: easter.py; Punkte: 8)

Implementieren Sie eine Funktion `easterdate`, welche eine Jahreszahl (Integer) als Argument erhält, und anschließend zu diesem Jahr das genaue Datum des Ostersonntags als Märzdatum (32. März = 1. April, usw.) ausgibt. Benutzen Sie dazu die ergänzte Osterformel nach Heiner Lichtenberg (https://de.wikipedia.org/wiki/Gau%C3%9Fsche_Osterformel#Eine_erg.C3.A4nzte_Osterformel). Testen Sie Ihre Funktion an geeigneten Beispielen, also z.B. mit:

```
>>> easterdate(2016)  
27. März  
>>> easterdate(2017)  
47. März
```

Lösung:

```
def easterdate(year):
    k = year // 100
    m = 15 + (3 * k + 3) // 4 - (8 * k + 13) // 25
    s = 2 - (3 * k + 3) // 4
    a = year % 19
    d = (19 * a + m) % 30
    r = (d + a // 11) // 29
    og = 21 + d - r
    sz = 7 - (year + year // 4 + s) % 7
    oe = 7 - (og - sz) % 7
    os = og + oe
    print(str(os) + ". März")
```

Aufgabe 3.3 (Erfahrungen; Datei: `erfahrungen.txt`; Punkte: 2)

Legen Sie im Unterverzeichnis `sheet03` eine Textdatei `erfahrungen.txt` an. Notieren Sie in dieser Datei kurz Ihre Erfahrungen beim Bearbeiten der Übungsaufgaben (Probleme, Bezug zur Vorlesung, Interessantes, benötigter Zeitaufwand, etc.).