

# Unit 8

Dr. Günter Kolousek

21. Juli 2015

Lege wiederum ein Verzeichnis an. Nennes es `11_unit8`! In diesem Verzeichnis sollen alle Dateien der jeweiligen Einheit abgelegt werden.

## 1 Schulübungen

1. Schreibe ein Programm `sumdigits.py`, das den Benutzer nach einer Zahl fragt und danach die Ziffernsumme ausgibt!

Verwende direkt den String, der von `input()` zurückkommt, um auf die einzelnen Ziffern zuzugreifen!

2. Baue das Programm `sumdigits.py` so um, dass es auch mit negativen Zahlen zurechtkommt. Welche Möglichkeiten gibt es?
3. Baue das Programm `sumdigits.py` so um, dass es auch mit Gleitkommazahlen zurechtkommt.
4. Schreibe ein Programm `power8.py`, das den Benutzer nach einer Zahl fragt, diese Zahl als eine Zahl im Oktalsystem interpretiert und diese mit der Potenzmethode in das Dezimalsystem umwandelt und ausgibt!
5. Erweitere das Programm `power8.py` so, dass außerdem überprüft wird, ob es sich um eine korrekte Zahl handelt und im Fehlerfall eine entsprechende Ausgabe tätigt.
6. Schreibe ein Programm `horner8.py`, das den Benutzer nach einer Zahl fragt, diese Zahl als eine Zahl im Oktalsystem interpretiert und diese mit der Hornermethode in das Dezimalsystem umwandelt und ausgibt!  
  
Überprüfe auch gleich, ob es sich um eine korrekte Zahl handelt wie im vorhergehenden Beispiel.
7. Schreibe ein Programm `arithmean.py`, das den Benutzer nach genau 10 Zahlen fragt und das arithmetische Mittel ausgibt.

8. Man kann die Quadratwurzel näherungsweise folgendermaßen bestimmen:

$$\begin{aligned}x_1 &= (x_0 + a/x_0)/2 \\x_2 &= (x_1 + a/x_1)/2 \\x_3 &= (x_2 + a/x_2)/2 \\&\dots \\x_n &= (x_{n-1} + a/x_{n-1})/2\end{aligned}$$

$x_0$  und  $a$  sind jeweils mit der Zahl zu initialisieren, von der die Quadratwurzel bestimmt werden soll. Bei  $x_n$  handelt es sich um die errechnete Näherung.

Schreibe ein Programm `squareroot.py`, das eine Zahl erfragt, von der die Quadratwurzel bestimmt werden soll und weiters nach einer Anzahl von Iterationsschritten (eine Iteration ist ein Durchgang einer Schleife). Diese Anzahl von Iterationsschritten ist in den obigen Formeln mit  $n$  bezeichnet.

Danach soll sowohl die errechnete Näherung als auch die genaue Lösung ausgegeben werden. Weiters soll der Fehler als Absolutbetrag (Funktion `abs`) der Differenz von der Konstanten `pi` aus dem Modul `math` und dem errechneten Näherungswert ausgegeben werden.

Verwende entsprechende Namen!

9. Schreibe ein Programm `regular_polygon2.py`, das ein regelmäßiges Vieleck mit einer beliebigen Anzahl von farbigen Seiten zeichnen kann.

- a) Schreibe dazu eine Funktion `regular_polygon1(length, colors)`, die die Seitenlänge `length` und ein Tupel mit Farbwerten als Strings im Parameter `colors` bekommt und das Vieleck vom Koordinatenursprung aus mit der gegebenen Seitenlänge und den angegebenen Farben zeichnet. Die erste Seite liegt in der positiven x-Achse. Das Polygon wird gegen den Uhrzeigersinn gezeichnet.

D.h. wird die Funktion folgendermaßen aufgerufen

```
regular_polygon(30, ("red", "blue", "green"))
```

dann wird ein gleichseitiges Dreieck mit der Kantenlänge 30 gezeichnet, wobei die erste Seite die Farbe rot, die zweite Seite blau und die dritte Seite grün hat.

- b) Erweitere `regular_polygon2.py`, sodass der Benutzer nach der Seitenlänge, nach der Anzahl der Seiten und danach nach den Farben gefragt wird.

## 2 Hausübung

1. Kap 8 durchlesen!
2. 2er Potenzen bis 16 in Binär- und Hexrepräsentation auswendig lernen!