

Einführung in die Informatik

WiSe 2025/2026

Hausaufgabenblatt 02, Abgabe am 10.11.25 um 12:00 Uhr

Allgemeine Hinweise

- Wenn die Hausaufgabe exakte Benennungen von Dateien, Funktionen, Variablen, etc. vorgibt, dann **müssen** Sie sich an diese Vorgabe halten. Falls nicht wird Ihre Abgabe nicht oder nur mit Punktabzügen korrigiert.
- Gleiches gilt, wenn die Aufgabe konkrete Anweisungen für `input()` oder `print()` vorgibt.
- Jede gefragte Ausgabe muss mit dem Befehl `print()` ausgegeben werden. Sofern nicht anders vorgegeben, geben Sie nur das geforderte Ergebnis und **keinen** zusätzlichen Text aus.
- Benutzen Sie **keine Python-Imports**, außer dies ist in der Aufgabenstellung explizit erlaubt.
- Beachten Sie die Anweisungen, die ggf. zusätzlich in den jeweiligen Aufgaben gegeben sind.
- Geben Sie für jede Aufgabe eine separate Datei ab. Benennen Sie diese Datei als `bXXaY.py`, wobei XX die Blattnummer und Y die Aufgabennummer ist.
- Schreiben Sie die Namen aller Gruppenmitglieder als Kommentar in die erste Zeile jeder Datei.
- Programmcode muss möglichst einfach und gut lesbar sein, um die Korrektur zu erleichtern. Code, der nicht oder nur schwer lesbar ist, kann zu Punktabzug führen.

Hausaufgabe 1 (2 Punkte):

Schreiben Sie ein Programm, welches eine Ganzzahl n einliest und die Anzahl der *echten* Teiler von n berechnet und ausgibt. Sie dürfen davon ausgehen, dass tatsächlich eine Ganzzahl eingegeben wird.

Hinweis: Eine Zahl x ist ein echter Teiler von n , wenn $x \neq 1$, $x \neq n$, sowie x teilt n gilt.

Hausaufgabe 2 (2 Punkte):

Schreiben Sie ein Programm, welches einen String einliest und, mit Komma getrennt, die Zahlenwerte der Zeichen ausgibt.

Hinweis: Nutzen sie `ord(zeichen)` um den Zahlenwert eines Zeichens zu ermitteln.

Beispiel: Für die Eingabe ASCII soll "65,83,67,73,73" ausgegeben werden.

Hausaufgabe 3 (4 Punkte):

Schreiben Sie ein Programm, das überprüft, ob ein eingegebener String eine Ganzzahl ohne führende 0en darstellt. Sie dürfen **nicht** `try` und `except` oder sonstige Befehle zum Abfangen von Exceptions verwenden.

Beispiele:

"0"	→ True	"00"	→ False	"100"	→ True	"-100"	→ True
"--100"	→ False	"abc"	→ False	" 10 "	→ False	"1-0"	→ False
"01"	→ False	"-01"	→ False	"-0"	→ True	"-00"	→ False

Hausaufgabe 4 (2 Punkte):

Schreiben Sie ein Programm, welches vom Nutzer eine Zeichenkette und ein Zeichen einliest und die erste Position des Zeichens in der Zeichenkette ausgibt, falls diese existiert. Ansonsten soll -1 ausgegeben werden. Hierbei ist, wie gewöhnlich, die Position des ersten Zeichens die Position 0. Sie dürfen davon ausgehen, dass die zweite Eingabe tatsächlich ein einzelnes Zeichen, d.h. eine Zeichenkette der Länge 1, ist.

Hausaufgabe 5 (7 Punkte):

Schreiben Sie ein Programm, welches eine Zeichenkette w bestehend aus mehreren Wörtern, getrennt durch Leerzeichen, einliest.

Bestimmen Sie dann für jedes Wort die größte aufeinanderfolgende Anzahl an Konsonanten und geben Sie diese zusammen mit dem Wort aus. Sie dürfen davon ausgehen, dass nur das lateinische Alphabet verwendet und die Wörter korrekt getrennt wurden.

Hinweis: Konsonanten sind alle Buchstaben außer a,e,i,o,u, wobei Groß- und Kleinschreibung keine Rolle spielt.

Für den eingegebenen Satz Technik macht Spass würde die Ausgabe also wie folgt lauten:

Technik 3

macht 3

Spass 2

Hausaufgabe 6 (3 Punkte):

Schreiben Sie ein Programm, welches zwei positive Ganzzahlen einliest. Hierbei steht der erste eingelesene Wert für die Höhe einer Matrix, d.h. die Anzahl der Zeilen und der zweite eingelesene Wert für die Breite einer Matrix, d.h. die Anzahl der Spalten. Ausgegeben werden soll die Matrix mit den eingegebenen Dimensionen, welche sich aus der größtmöglichen Einheitsmatrix und 0 in allen restlichen Spalten bzw. Zeilen zusammensetzt.

Hinweis: Die n -Einheitsmatrix ist die Matrix der Größe $n \times n$ in welcher auf der fallenden Diagonalen alle Einträge 1 und alle restlichen Einträge 0 sind.

Beispiele: Die nachfolgenden Ein-/Ausgaben sind korrekt:

4	3	5	2
4	5	3	1
1000	10000	100	1
0100	01000	010	0
0010	00100	001	
0001		000	
		000	

Sie dürfen davon ausgehen, dass tatsächlich zwei positive Ganzzahlen eingegeben werden.

Präsenzaufgaben

Präsenzaufgabe 1:

Schreiben Sie ein Programm, welches einen String x einliest und anschließend alle Teilstrings von x , durch Kommas getrennt, ausgibt. Dabei sollen die Teilstrings sortiert nach ihrer Länge und ihrem Startpunkt ausgegeben werden, d.h. zuerst alle Teilstrings der Länge 1, dann alle Teilstrings der Länge 2, usw. Die Teilstrings gleicher Länge sollen nach ihrem Startpunkt sortiert sein, d.h. der Teilstring beginnend bei Index 0 kommt vor dem beginnend bei Index 1, usw.

Beispiele:

```
> 01234
0,1,2,3,4,01,12,23,34,012,123,234,0123,1234,01234

> abbc
a,b,b,c,ab,bb,bc,abb,bbc,abbc
```

Präsenzaufgabe 2:

Geben Sie für jede der folgenden Bedingungen einen Python-Ausdruck an, der genau dann `True` ist, wenn ein String x die Bedingung erfüllt. Finden Sie außerdem einen möglichst kurzen String, welcher alle drei Bedingungen erfüllt.

1. Der String x ist ein Palindrom.

Beispiele:

```
"lagerregal"  → True      "lagerhalle" → False
```

2. Jedes Zeichen in x ist gedoppelt.

Beispiele:

```
"aabbaacc"  → True      "bbaacaca"  → False
```

3. Die erste Hälfte von x ist lexikographisch echt-kleiner als die zweite Hälfte. Wenn x eine ungerade Länge hat, soll das Zeichen in der Mitte ignoriert werden.

Beispiele:

```
"abermals"  → "aber" < "mals"  → True
"aufgabe"   → "auf"  > "abe"   → False
```

Präsenzaufgabe 3:

Diskutieren Sie, welche der folgenden Schleifen terminieren:

```
i=1
j=20
while i+j<100:
    i=i+2
    j=j-1
```

```
i=1
j=3
while i+j<100:
    i=i+1
    j=j-1
```

```
i=100
j=27
while i!=j:
    i=i/2
    j=j/3
```

Testen Sie Ihre Vermutungen, soweit möglich.

Präsenzaufgabe 4:

Schreiben Sie ein Programm, welches einen Float von der Standardeingabe einliest. Dieser gibt eine Temperatur in °C an und soll von dem Programm nach Kelvin und °F umgerechnet werden. Nutzen sie dafür die Umrechnungen

$$\text{Kelvin} = \text{Celsius} + 273.15 \text{ und } \text{Fahrenheit} = \frac{\text{Celsius} \cdot 9}{5} + 32.$$

Ausgegeben werden sollen alle drei Temperaturangaben, gerundet auf 3 Nachkommastellen. Hierbei sollen Werte die auf 5 enden immer aufgerundet werden. Sie dürfen davon ausgehen, dass tatsächlich ein Float-Wert eingegeben wird.