

PROPÄDEUTIKUM INFORMATIK

SOSE 2018

Martin Mehlhose

INFORMATIONSVERANSTALTUNG

- Informationsveranstaltung zur Bewerbung und Studium an der Universität Leipzig und TU Dresden
- **09. Mai 15:00 Uhr** in der Aula (UL)
- **22. Mai 13:30 uhr** im Zimmer 27 (TUD)

LANGE NACHT DER COMPUTERSPIELE

- Samstag 11.05.2019 ab 14 Uhr
- HTWK Leipzig
- <https://computerspielenacht.htwk-leipzig.de/computerspielenacht/>

LISTEN

LISTEN

- Listen dienen dem "effizientem" Speichern großer Datenmengen
- Listen werden durch den [] Operator markiert. Zum Beispiel [1,2,3,4,5,6,7]
- Die Daten müssen nicht alle den selben Typ haben

LISTEN

- Zugriff auf Listen Elemente über den Index.
Achtung Listen sind 0 basiert!
- Zugriff erfolgt dann über eckige Klammern, z.B.
list[0] für das erste Listenelement
- Die Elemente können über den Index sowohl
ausgelesen als auch bearbeitet werden
- sollen alle Elemente einer Liste bearbeitet werden,
dass am besten eine for-Schleife benutzen

LISTEN

- Wichtige Funktionen:
- `len(list)` : gibt sie Länge der Liste zurück
- `mylist.pop(i)` : entfernt das i'te Element aus der Liste
- `mylist.append(i)` : hängt i an das Ende der Liste an
- Zusammenfügen von zwei Listen über + Operator

LISTEN

- Achtung: Listen sind Referenzdatentypen
- -> wir können Listen nicht einfach kopieren
- -> in Funktionen geänderte Listen brauchen nicht über return zurückgegeben werden

- Vorsicht beim kopieren von Listen !

STRINGS

STRINGS

- String ist ein Datentyp der Zeichenketten speichern kann
- Einer der wichtigsten Datentypen
- Anlegen eines Strings durch ""

FORMATIERUNG

- Einfache Formatierungsaufgaben können durch Steuerzeichen erledigt werden
- Einrücken um einen Tabulator : `\t`
- Zeilenumbruch: `\n`
- Soll in einem String ein " ausgegeben werden, muss dies \" geschrieben werden
- Soll ein \ geschrieben werden, so muss/sollte dies \\ geschrieben werden

WICHTIGE FUNKTIONEN

- String build-in Funktionen werden immer über Punktnotation aufgerufen
- `.count("xy")` liefert die Anzahl der Vorkommen von "xy" im String
- `.find("xy")` liefert den ersten Index im String, an dem "xy" steht
- `.split("xy")` trennt den String an den Stellen xy und liefert eine Liste der einzelnen Komponenten
- `.replace("alt", "neu")` ersetzt alle vorkommen von "alt" in "neu"

WICHTIGE FUNKTIONEN

- `.lower()` liefert den String in Kleinbuchstaben
- `.upper()` liefert den String in Großbuchstaben
- `.isDigit()` liefert `True`, wenn der String nur aus Zahlen besteht

MASCHINENZAHLEN

MASCHINENZAHLEN

- Wir rechnen hier mit 8 bit Zahlen
- Also Zahlen im Bereich von -128 bis + 127
- Umwandlung Dez in Bin durch iteratives teilen durch 2 und merken der Reste

MASCHINENZAHLEN

- Umrechnung von Bin nach Dec durch Multiplikation der letzten 7 bit mit ihrer jeweiligen Wertigkeit
- Beispiel: 0 0 0 0 1 0 1 0

MASCHINENZAHLEN

- Umrechnung von Bin nach Dec durch Multiplikation der letzten 7 bit mit ihrer jeweiligen Wertigkeit
- Beispiel: 0 0 0 0 1 0 1 0
- Lösung: $0 \cdot 2^0 + 1 \cdot 2^1 + 0 \cdot 2^2 + 1 \cdot 2^3 = 2 + 8 = 10$

MASCHINENZAHLEN

- Negative Zahlen
- Umrechnung des Betrags ins Binärsystem
- jedes Bit invertieren
- + 1 rechnen
- Beispiel: $10 = 0\ 0\ 0\ 0\ 1\ 0\ 1\ 0$
- Jedes Bit invertieren: $1\ 1\ 1\ 1\ 0\ 1\ 0\ 1$
- 1 addieren: $1\ 1\ 1\ 1\ 0\ 1\ 1\ 0 = -10$