

Junioraufgabe 1: Bücherregal

Team-ID: 12345

Team-Name: MyTeam

Bearbeiter/-innen dieser Aufgabe: Lena Müller, Malte Riedberg

Datum: 10. November 2017

Lösungsidee

Die Bücher werden nach Höhen sortiert in einer Liste gespeichert. Zu Beginn ist das erste, also das kleinste Buch, der Beginn des ersten Abschnitts. Dann gehen wir die Bücherliste durch. Sobald ein Buch gefunden wurde, das eine Höhendifferenz größer als 3 cm zum Beginn des Abschnitts hat, setzen wir vor diesem Buch eine Figur und nehmen dieses Buch als Beginn eines neuen Abschnitts. Das machen wir bis zum Ende der Bücherliste und zählen dabei die Anzahl der notwendigen Figuren. Wenn diese Anzahl kleiner oder gleich der Anzahl der verfügbaren Figuren ist, ist eine Aufstellung möglich und wir geben die Lösung aus.

Umsetzung

Die Lösungsidee wird in ein Programm der Sprache Python umgesetzt. Die Höhen der Bücher sind in mm angegeben. Wir lesen die Höhen in eine Liste buecher ein:

```
In [9]: eingabe = 'buecherregal1.txt'
f = open('beispieldaten/'+eingabe)

anz_figuren = int(f.readline())
anz_buecher = int(f.readline())

buecher = []          # Liste mit den Büchern

for i in range(anz_buecher):
    buecher.append(int(f.readline()))
f.close()

print(f'Beispieldatei: {eingabe}')
print(f'Anzahl Figuren: {anz_figuren}')
print(f'Anzahl Bücher: {anz_buecher}')
```

Beispieldatei: buecherregal1.txt

Anzahl Figuren: 4

Anzahl Bücher: 11

Dann sortieren wir die Bücherliste aufsteigend nach ihrer Höhe

```
In [10]: buecher.sort()
print(f'Buecher: {buecher}')
```

Buecher: [168, 170, 202, 211, 229, 233, 254, 260, 272, 306, 307]

Die Liste regal ist zu Beginn leer. Wir speichern darin der Reihe nach die Bücher und Figuren für die Aufstellung. Die Variable i zeigt auf den Index des Buches, das sich am Beginn eines Abschnitts befindet. Mit der Variablen j durchlaufen wir die Bücherliste.

Innerhalb der Schleife, mit der wir die Bücherliste durchlaufen, fügen wir ein Buch entweder mit append in die Liste regal ein, oder, falls das Buch zu groß ist, fügen wir eine Figur ins regal ein und dann das Buch. Die Anzahl der gesetzten Figuren merken wir uns in der Variablen zaehl. Mit i = j wird dann der Beginn des neuen Abschnitts gesetzt.

```
In [42]: regal = []          # die Aufstellung von Büchern und Figuren
        zaehl = 0           # Anzahl aufgestellter Figuren

        i = 0              # Anfang des Abschnitts
        j = 0              # aktuelles Buch
        while j < len(buecher): # solange noch nicht am Ende der Bücherliste
            if buecher[j] - buecher[i] <= 30: # passt aktuelles Buch noch in den Abschnitt?
                regal.append(buecher[j])      # Buch in den Abschnitt
            else:
                regal.append('Figur')         # Abschnittsende, setzen der Figur
                regal.append(buecher[j])     # aktuelles Buch ist Beginn des nächsten Abschnitts
                zaehl += 1                   # Erhöhe Anzahl aufgestellter Figuren
                i = j                       # Anfang des Abschnitts wird das aktuelle Buch
                j = j + 1                   # ein Buch weiter in der Liste

        print(f'Anzahl benötigter Figuren: {zaehl}')
```

Anzahl benötigter Figuren: 4

Falls die Anzahl benötigter Figuren größer als die verfügbare Anzahl ist, geben wir aus, dass eine Aufteilung nicht möglich ist. In der Ausgabe der Lösung erscheinen nur die Figuren, die für die Aufstellung notwendig sind. Falls weitere Figuren verfügbar sind, können diese beliebig platziert werden.

```
In [31]: if zaehl <= anz_figuren:          # Anzahl verwendeter Figuren <= Anzahl verfügbare Figuren?
        print(f'Aufteilung mit {anz_figuren} Figuren ist möglich:')
        print(regal)

        else:
            print(f'Aufteilung mit {anz_figuren} Figuren ist nicht möglich.')
```

Aufteilung mit 4 Figuren ist möglich:

[168, 170, 'Figur', 202, 211, 229, 'Figur', 233, 254, 260, 'Figur', 272, 'Figur', 306, 307]

Um das Ergebnis etwas übersichtlicher zu gestalten, lassen wir jeden Abschnitt in einer neuen Zeile beginnen.

```
In [32]: if zaehl <= anz_figuren:           # Anzahl verwendeter Figuren <= Anzahl verfügbare Figuren?
        print(f'Aufteilung mit {anz_figuren} Figuren ist möglich:')
        for x in regal:
            if x == 'Figur':
                print('\nFigur', end= ' ')
            else:
                print(x, end=' ')

        else:
            print(f'Aufteilung mit {anz_figuren} Figuren ist nicht möglich.')
```

Aufteilung mit 4 Figuren ist möglich:

168 170

Figur 202 211 229

Figur 233 254 260

Figur 272

Figur 306 307

Beispiele

Unser Programm liefert für die Beispieldaten folgende Ergebnisse:

Beispieldatei: buecherregal1.txt

Anzahl Figuren: 4

Anzahl Bücher: 11

Aufteilung mit 4 Figuren ist möglich:

168 170

Figur 202 211 229

Figur 233 254 260

Figur 272

Figur 306 307

Beispieldatei: buecherregal2.txt

Anzahl Figuren: 2

Anzahl Bücher: 7

Aufteilung mit 2 Figuren ist möglich:

169 175

Figur 203 209 210 229

Figur 235

Beispieldatei: buecherregal3.txt

Anzahl Figuren: 2

Anzahl Bücher: 7

Aufteilung mit 2 Figuren ist nicht möglich.

Beispieldatei: buecherregal4.txt

Anzahl Figuren: 4

Anzahl Bücher: 100

Aufteilung mit 4 Figuren ist möglich:

160 160 161 161 162 165 165 166 167 167 167 169 170 170 171 173 173 174 174 177 180 182 183 184 184 185 185
187 188 189 190

Figur 196 197 197 199 200 201 202 206 207 207 211 212 212 214 215 216 217 218 219 224 225

Figur 233 235 237 238 238 239 240 240 240 245 246 246 247 253 254 256 258 259 259 261

Figur 264 266 266 267 268 270 270 272 274 275 276 277 278 279 286 286 287 288 289 290 293 293
Figur 295 296 300 301 303 304

Beispieldatei: buecherregal5.txt

Anzahl Figuren: 3

Anzahl Bücher: 100

Aufteilung mit 3 Figuren ist möglich:

160 161 161 161 162 162 162 163 163 164 164 164 164 164 165 165 165 166 167 167 168 168 168 168 169 169 170

170 171 171 171 171 172 174 174 174 174 175 175 176 176 176 176 176 177 177 178 179 180 180

Figur 201 202 202 202 203 206 206 208 209 210 211 212 216 220 220 221 221 229 230 230 230 231

Figur 232 232 233 233 235 237 238 240 241 241 241 243 243 246 248 248 250 254 257 258 260 261 261

Figur 263 264 265 265 270

Beispieldatei: buecherregal6.txt

Anzahl Figuren: 4

Anzahl Bücher: 100

Aufteilung mit 4 Figuren ist nicht möglich.

Quellcode

```
In [ ]: eingabe = 'buecherregal1.txt'
f = open('beispieldaten/'+eingabe)

anz_figuren = int(f.readline())
anz_buecher = int(f.readline())

buecher = []          # Liste mit den Büchern

for i in range(anz_buecher):
    buecher.append(int(f.readline()))
f.close()

print(f'Beispieldatei: {eingabe}')
print(f'Anzahl Figuren: {anz_figuren}')
print(f'Anzahl Bücher: {anz_buecher}')

buecher.sort()

regal = []            # die Aufstellung von Büchern und Figuren
zaehl = 0             # Anzahl aufgestellter Figuren

i = 0                 # Anfang des Abschnitts
j = 0                 # aktuelles Buch
while j < len(buecher): # solange noch nicht am Ende der Bücherliste
    if buecher[j] - buecher[i] <= 30: # passt aktuelles Buch noch in den Abschnitt?
        regal.append(buecher[j])      # Buch in den Abschnitt
    else:
        regal.append('Figur')          # Abschnittsende, setzen der Figur
        regal.append(buecher[j])       # aktuelles Buch ist Beginn des nächsten Abschnitts
        zaehl += 1                     # Erhöhe Anzahl aufgestellter Figuren
        i = j                          # Anfang des Abschnitts wird das aktuelle Buch
        j = j + 1                      # ein Buch weiter in der Liste

if zaehl <= anz_figuren:                # Anzahl verwendeter Figuren <= Anzahl verfügbare Figuren?
    print(f'Aufteilung mit {anz_figuren} Figuren ist möglich:')
    for x in regal:
```

```
    if x == 'Figur':  
        print('\nFigur', end= ' ')  
    else:  
        print(x, end=' ')  
  
else:  
    print(f'Aufteilung mit {anz_figuren} Figuren ist nicht möglich.')
```