

1. (3 Punkte) Beschreibe in Worten, welchen Wert die Funktion *doit* ermittelt. Was wird im angegebenen Beispiel ausgegeben?

```
def doit(a):
    best = None
    best_val = -float('inf')
    for i in range(len(a)-1):
        val = a[i+1]
        if val > best_val:
            best_val = val
            best = i
    return best

a = [50,32,66,94,42,14,77,89]
print(doit(a))
```

Lösung:

Die Funktion *doit* führt eine lineare Suche auf der Liste *a* durch. Sie ermittelt den Index der Zahl, die den größten Nachfolger hat. Die Ausgabe ist: 2

2. (3 Punkte) Um mit der lineare Suche das größte Element in einer Liste *a* zu finden, wird die folgende Funktion *doit* implementiert. Gib eine Liste *a* an, bei der ein falscher Wert zurückgegeben wird. Gib auch diesen falschen Wert an.

```
def doit(a):
    best = None
    best_val = 0
    for i in range(len(a)):
        val = a[i]
        if val > best_val:
            best_val = val
            best = i
    return best
```

Lösung:

a = [-4,-2,-9]

Es wird None zurückgegeben.

3. (4 Punkte) Die Aufgabe bezieht sich auf die Implementation der binaeren Suche aus dem Unterricht. Die binaere Suche besteht aus (in der Regel) mehreren Durchgängen, in denen immer wieder der linke oder rechte Zeigefinger (*i* und *j*) versetzt wird, bis in der Mitte die gesuchte Zahl gefunden wurde oder die Finger sich überkreuzt haben.

Beispiel: Bei Suche nach $x = 16$ in der Liste *a* = [4, 7, 12, 13, 19] stehen die Variablen *i*,*j*,*mitte* zu Beginn auf 0, 4, 2. Am Ende des ersten Durchgangs haben die Variablen die Werte auf 3, 4, 3.

```
x = 42
a = [4, 7, 10, 11, 13, 17, 21, 24, 27, 29, 30, 32, 40, 45]
binaereSuche(a,x)
```

Zu Beginn eines Durchgangs sollen die Variablen *i*,*j*,*mitte* die Werte 7 13 10 haben.

- a. Welche Werte haben sie am Ende des Durchgangs?
b. Wird ein weiterer Durchgang folgen?

Lösung: a. 11 13 12 b. Ja