

Aufgabe 1

Zeige schrittweise, wie die Zahlen in der Liste $L = [9, 3, 6, 8, 7]$ mit dem Selectionsort-Verfahren aufsteigend sortiert werden. Wie viele Vergleiche und Vertauschungen sind für jeden Schritt nötig?

Aufgabe 2

Zeige schrittweise, wie die Zahlen in der Liste $L = [9, 3, 6, 8, 7]$ mit dem Insertionsort-Verfahren aufsteigend sortiert werden. Wie viele Vergleiche und wie viele Arrayoperationen sind für jeden Schritt nötig?

Aufgabe 5

Zeige schrittweise, wie Elemente der Liste $L = [8, 7, 5, 4, 1]$ mit dem Bubblesort-Verfahren aufsteigend sortiert werden. Wie viele Vergleiche und Vertauschungen sind dafür insgesamt nötig?

Aufgabe 6

Wie viele Vergleiche und wie viele Vertauschungen benötigt Selectionsort für eine Liste mit 7 Elementen,

- (a) die aufsteigend sortiert ist,
- (b) die absteigend sortiert ist?

Aufgabe 7

Wie viele Vergleiche und wie viele Speicheroperationen benötigt Insertionsort für eine Liste mit 7 Elementen,

- (a) die aufsteigend sortiert ist,
- (b) die absteigend sortiert ist?

Aufgabe 8

Welcher Sortieralgorithmus wird mit dem folgenden Python-Programm implementiert?

```
1 def sort(A):
2     n = len(A)
3     for i in range(0, n-1):
4         k = i
5         for j in range(i+1, n):
6             if A[j] < A[k]:
7                 k = j
8         A[k], A[i] = A[i], A[k]
```

Aufgabe 9

Bestimme anhand der ersten 4 Verarbeitungsschritte, um welchen Sortieralgorithmus (Gnomesort, Selectionsort, Insertionsort, Bubblesort) es sich handelt.

Beachte:

- Nicht alle Algorithmen sortieren die Liste bis zum Ende.
- Es werden nur Schritte dargestellt, die *potenziell* Elemente vertauschen oder verschieben. Dabei kann es vorkommen, dass ein Element mit sich selbst vertauscht oder um null Positionen verschoben wird.
- Ein Algorithmus wird höchstens einmal verwendet.

(a)

4	5	1	3	2
4	5	1	3	2
1	4	5	3	2
1	3	4	5	2
1	2	3	4	5

(b)

4	5	1	3	2
1	5	4	3	2
1	2	4	3	5
1	2	3	4	5
1	2	3	4	5

Aufgabe 10

Gib die Sortierdauer der folgenden Sortieralgorithmen in der Big-Oh-Notation an.

	Worst Case	Best Case
Insertionsort		
Gnomesort		
Bubblesort		
Selectionsort		

Aufgabe 14

- (a) Zeige, wie der Partitionierungsschritt von Quicksort den folgenden Listenausschnitt verarbeitet. Protokolliere dein Vorgehen, indem du jeweils eine neue Zeile beginnst, wenn ein Element (evtl. auch an Ort und Stelle) getauscht wird.

...	2	8	3	9	4	6	...
...							...
...							...
...							...
...							...
...							...
...							...
...							...
...							...
...							...

- (b) Welchen Wert gibt der Partitionierungsschritt zurück wenn das erste Element der obigen Liste (also 2) den Index $i = 100$ hat?

Aufgabe 15

Beschreibe, für welche Art von Listen der Quicksort-Algorithmus die Worst Case-Laufzeit aufweist?

Aufgabe 16

Beschreibe zwei Pivotstrategien für das Quicksort-Verfahren, um die Worst Case-Laufzeit zu verhindern.

