

1. Du kannst schrittweise zeigen, wie Gnomesort eine Liste von Zahlen aufsteigend sortiert.

| | |
|----------------|----------------|
| <u>3</u> 1 4 2 | 1 3 <u>2</u> 4 |
| 3 <u>1</u> 4 2 | 1 <u>2</u> 3 4 |
| <u>1</u> 3 4 2 | 1 2 <u>3</u> 4 |
| 1 <u>3</u> 4 2 | 1 2 3 <u>4</u> |
| 1 3 <u>4</u> 2 | 1 2 3 4 _ |
| 1 3 4 <u>2</u> | |

2. Du kannst angeben, wie viele Schritte Gnomesort benötigt, um eine aufsteigend sortierte Liste mit n Elementen „nochmals“ aufsteigend zu sortieren.

| | | |
|------------------|------------------|--------------|
| <u>1</u> 2 3 4 5 | 1 2 3 <u>4</u> 5 | ⇒ 5 Schritte |
| 1 <u>2</u> 3 4 5 | 1 2 3 4 <u>5</u> | |
| 1 2 <u>3</u> 4 5 | 1 2 3 4 5 _ | |

Verallgemeinerung: n Elemente ⇒ n Schritte.

3. Du kannst angeben, wie viele Schritte Gnomesort benötigt, um eine absteigend sortierte Liste mit n Elementen aufsteigend zu sortieren.

| | | |
|--------------|--------------|--------------|
| <u>3</u> 2 1 | 2 <u>1</u> 3 | ⇒ 9 Schritte |
| 3 <u>2</u> 1 | <u>1</u> 2 3 | |
| <u>2</u> 3 1 | 1 <u>2</u> 3 | |
| 2 <u>3</u> 1 | 1 2 <u>3</u> | |
| 2 3 <u>1</u> | 1 2 3 _ | |

Verallgemeinerung: n Elemente ⇒ n^2 Schritte.

4. Du kannst den Python-Code angeben, der in einer Liste L die Elemente mit den Indizes i und j vertauscht.

- *Lösung 1:* (mit Mehrfachzuweisung)

```
L[i], L[j] = L[j], L[i]
```

- *Lösung 2:* (mit Hilfsvariable)

```
tmp = L[i] # i-tes Element in Hilfsvariable tmp speichern
```

```
L[i] = L[j] # L[i] mit L[j] überschreiben
```

```
L[j] = tmp # L[j] mit tmp überschreiben
```