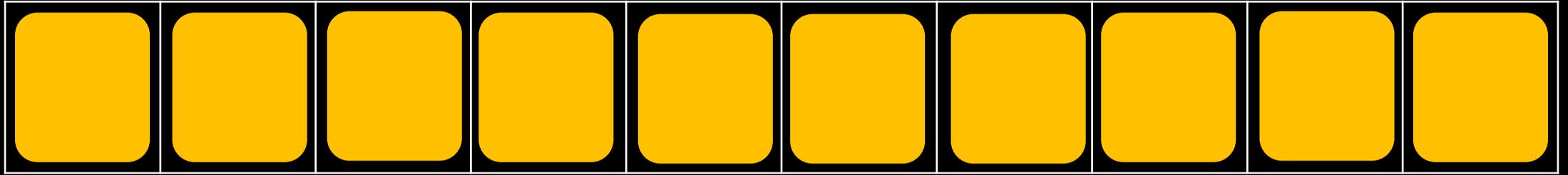


Informatik Q1 Abels



Lineare Suche

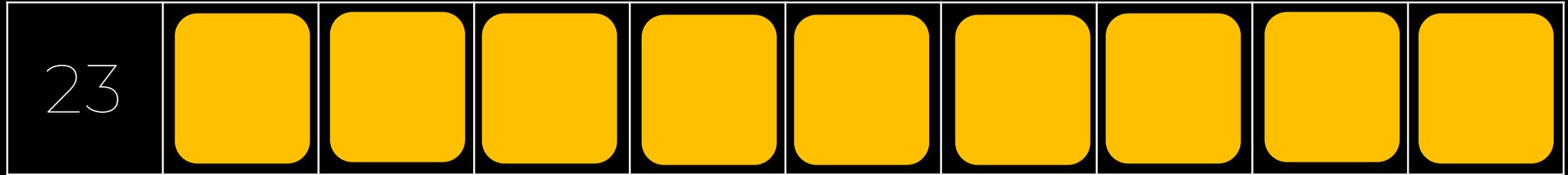
Lineare Suche



Wo ist die 29 ?



Lineare Suche



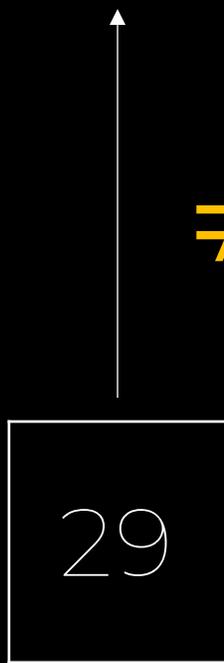
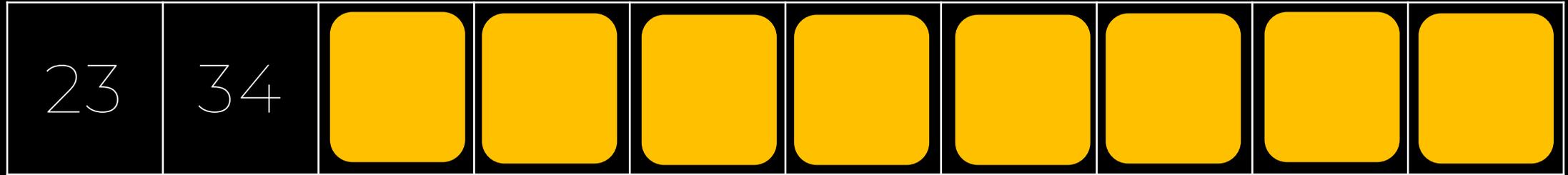
≠

29



... nicht an 0. Stelle ...

Lineare Suche



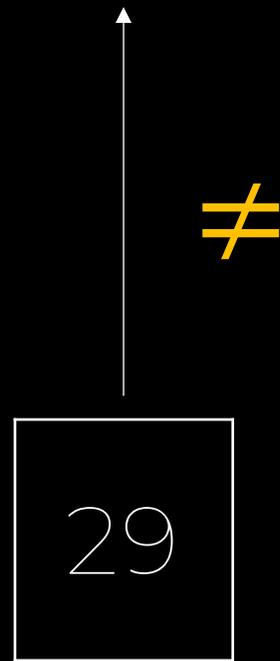
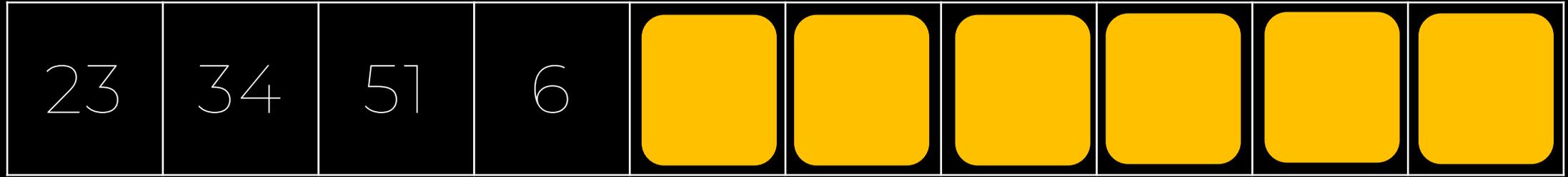
... nicht an 1. Stelle ...

Lineare Suche



... nicht an 2. Stelle ...

Lineare Suche



... nicht an 3. Stelle ...

Lineare Suche



≠

... nicht an 4. Stelle ...

29



Lineare Suche

23	34	51	6	17	85				
----	----	----	---	----	----	--	--	--	--

≠

... nicht an 5. Stelle ...

29



Lineare Suche

23	34	51	6	17	85	29			
----	----	----	---	----	----	----	--	--	--

... an 6. Stelle!

29

=





Übung 1

- Formuliere in deinen eigenen Worten einen Algorithmus, ein Element in einer Liste nach dieser Strategie zu suchen.
- Analysiere die Laufzeit des Algorithmus: Wie viele Schritte benötigst du im Worst-Case?
- Zeichne zu deinem Algorithmus ein Struktogramm.
- Implementiere deinen Algorithmus. Erstelle dazu ein Programm namens **LinearSearch**, in dem das Element **e (int)** in der Liste **liste** gesucht wird. Die Stelle, an der das Element gefunden wurde, soll im Terminal ausgegeben werden.
- Erweitere dein Programm, sodass auch der Fall berücksichtigt wird, dass das Element nicht im Array vorhanden ist.
- Wiederhole Teil c), dieses Mal jedoch mit einem anderen Schleifen-Typ.

```
LinearSearch.py
1 liste = [3, 42, 1, -5, 10, 8]
2 e = 42
3
4 # DEIN CODE
```



```
Terminal
1
```

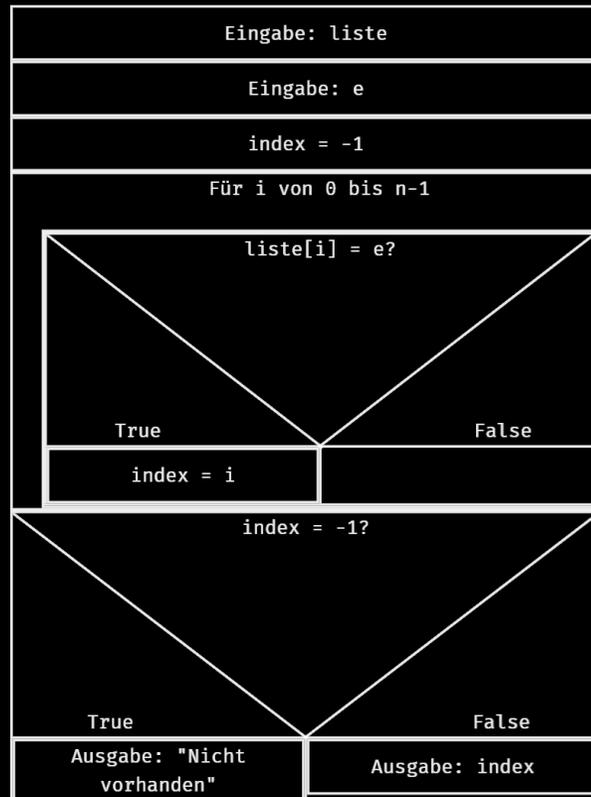




Übung 1

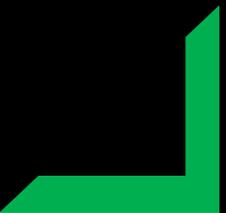


Laufzeit: **$O(n)$** , denn im Worst-Case durchläuft der Algorithmus die ganze Liste.



LinearSearch.py

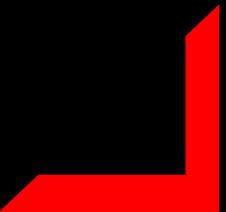
```
1 liste = [3, 42, 1, -5, 10, 8]
2 e = 42
3
4 index = -1
5 for i in range(0, len(liste)-1):
6     if liste[i] == e:
7         index = i
8 if index == -1:
9     print("Nicht vorhanden")
10 else:
11     print("Index:", index)
```





Tagebucheintrag

Lineare Suche





Wochenübung

Schreibe ein Programm **SuchenUndSortieren**, das die Funktion **linear_search** implementiert und mit den 2 relevanten Testcases prüft.

```
SuchenUndSortieren.py

1 def linear_search(liste, e):
2     pass
3
4 print("Testcase 1")
5 # ...
```

Terminal

```
Testcase 1
Liste: [3, 42, 1, -5, 10, 8]
Element: 42
Index: 1

-----
Testcase 2
Liste: [3, 42, 1, -5, 10, 8]
Element: 4
Index: Nicht vorhanden

-----
```