

# Informatik Q1 Abels



Was ist eine **Lineare Suche**?

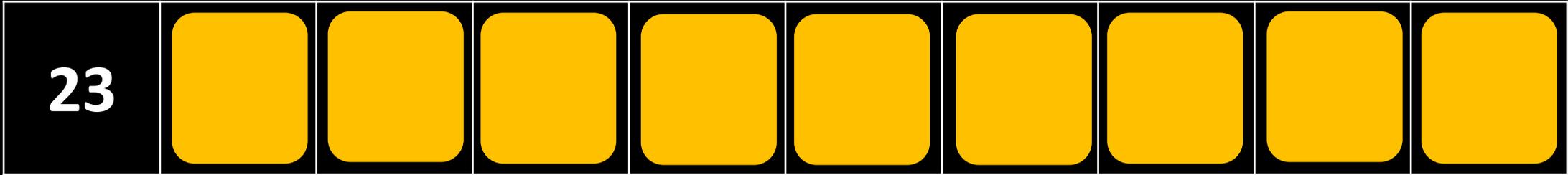


Wo ist die

29

?



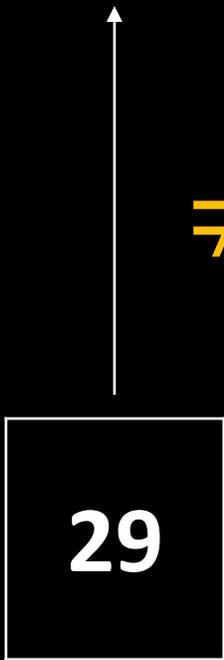
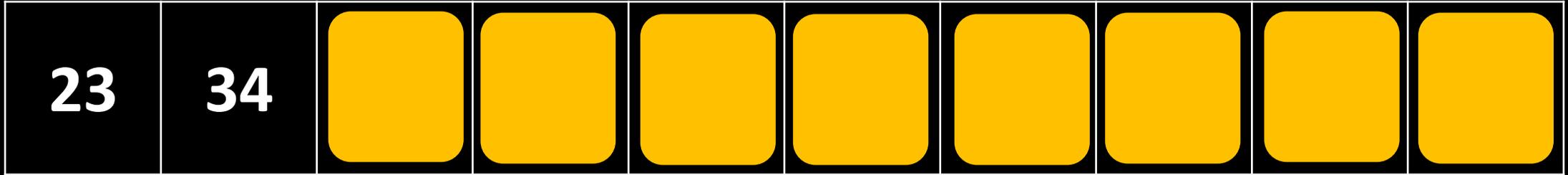


≠

29



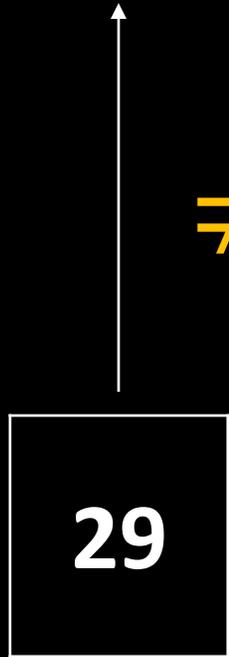
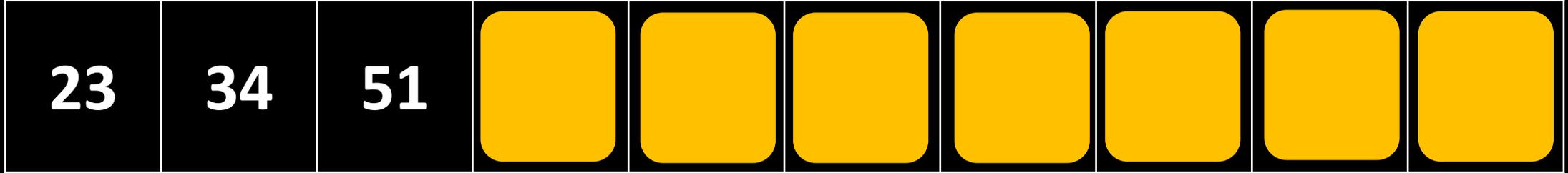
... nicht an 0. Stelle ...



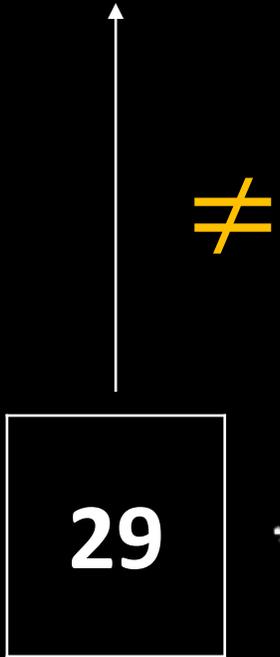
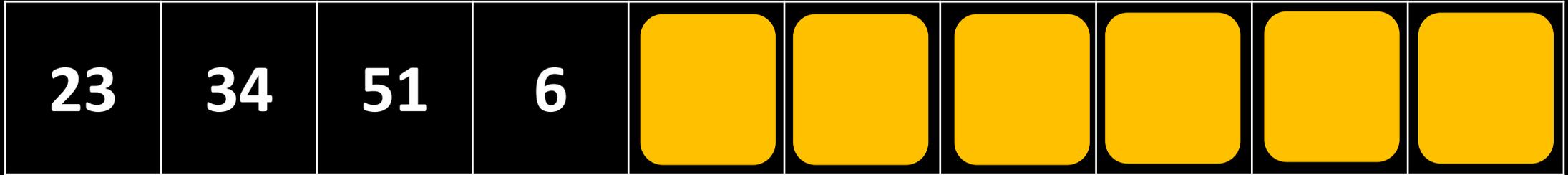
≠



... nicht an 1. Stelle ...



... nicht an 2ter Stelle ...



... nicht an 3. Stelle ..



≠



... nicht an 4. Stelle ...



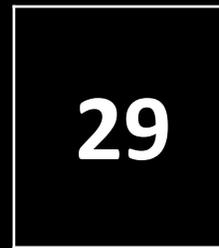
≠



... nicht an 5. Stelle ...



... an 6. Stelle!



=





# Übung 1

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



# Tagebucheintrag



Lineare Suche



# Wochenübung

Schreibe ein Programm **SuchenUndSortieren** mit der Funktion

```
int linearSearch(int e, int[] list)
```

, die ein Element und einen Array übergeben bekommt und den Index des gefundenen Elements oder -1 zurückgibt. In der **main**-Methode sollen alle nötigen Testcases (2!) getestet werden.