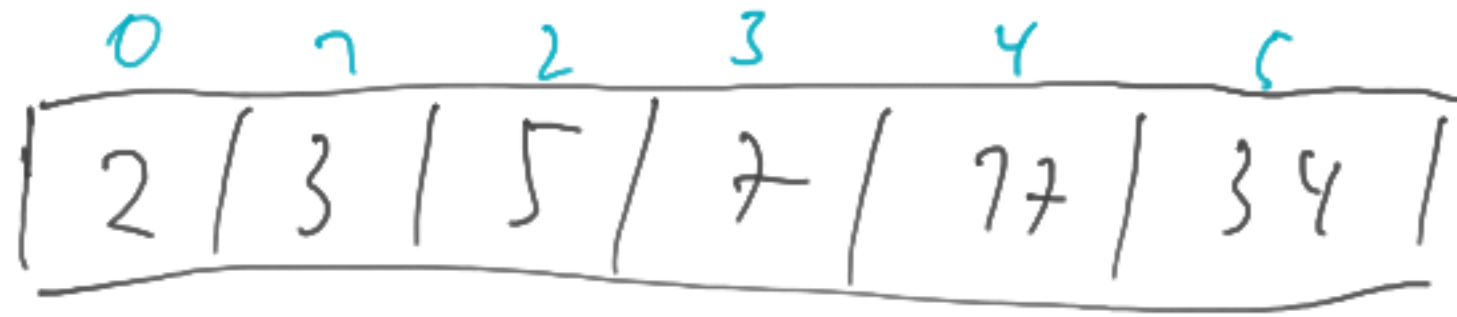


Lineare Suche



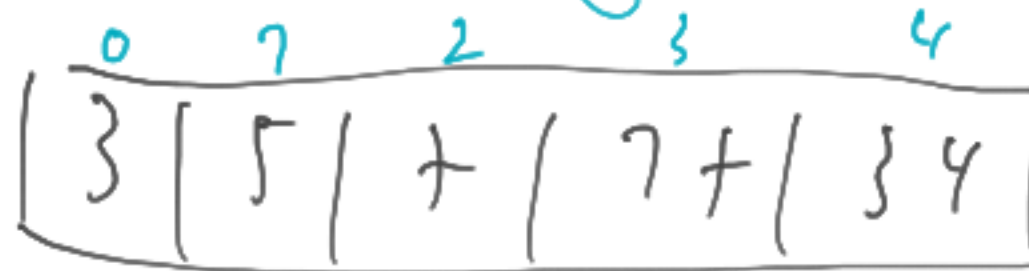
An welcher Stelle steht x?

Prinzip bei rekursiver linearer Suche:

Den Kopf der Liste getrennt vom Rest behandeln

Kopf: Rekursionsanfang

2



Wenn der Kopf == x ist:

return 0

Die Position des Kopfs.

Wenn der Kopf != x ist:

In der restlichen Liste weitersuchen.

Die gefundene Position +1 liefern.

Beispiel

17 ist in Rest-Liste an Stelle 3, in Original-Liste ist sie eins weiter.

$$\text{find}(\text{list}, x) = \begin{cases} 0, & \text{falls } \text{list}[0] == x \\ \text{find}(\text{list}[1:], x) + 1 & \text{sonst} \end{cases}$$

Noch nicht beachtet: Verhalten, wenn x nicht enthalten ist.

Binäre Suche

Annahme:
Liste ist
sortiert.

0	1	2	3	4	5
2	3	5	7	77	34

An welcher Stelle steht x?

1. x mit dem
mittleren
Element
vergleichen.

2. Wenn
gefunden,
dann Position
des mittleren
Elements
liefern.

2. Wenn nicht
gefunden,
dann rekursiv
links oder
rechts
weitersuchen.

Wenn m die
mittlere
Position ist ...

Muss in der
Funktion
berechnet
werden.

$m = \text{len}(\text{list}) / 2$

$\text{list}[:m]$

$\text{list}[m+1:] + m$

$\text{find}(\text{list}, x)$
 $= \begin{cases} m, & \text{falls } \text{list}[m] == x \\ \text{find}(\text{list}[:m], x), & \text{falls } x < \text{list}[m] \\ \text{find}(\text{list}[m+1:], x) + m, & \text{sonst} \end{cases}$

2	3	5	7	77	34
---	---	---	---	----	----

