

Grundsätzliches Aussehen
einer verketteten Liste



Schrittweiser Aufbau

Leere
Liste



ungültiges
Element

Dummy-
Element

markiert
das Ende
der Liste

Schrittweiser Aufbau

Anhängen eines neuen Elements

Ein
Element



gültiges
Element

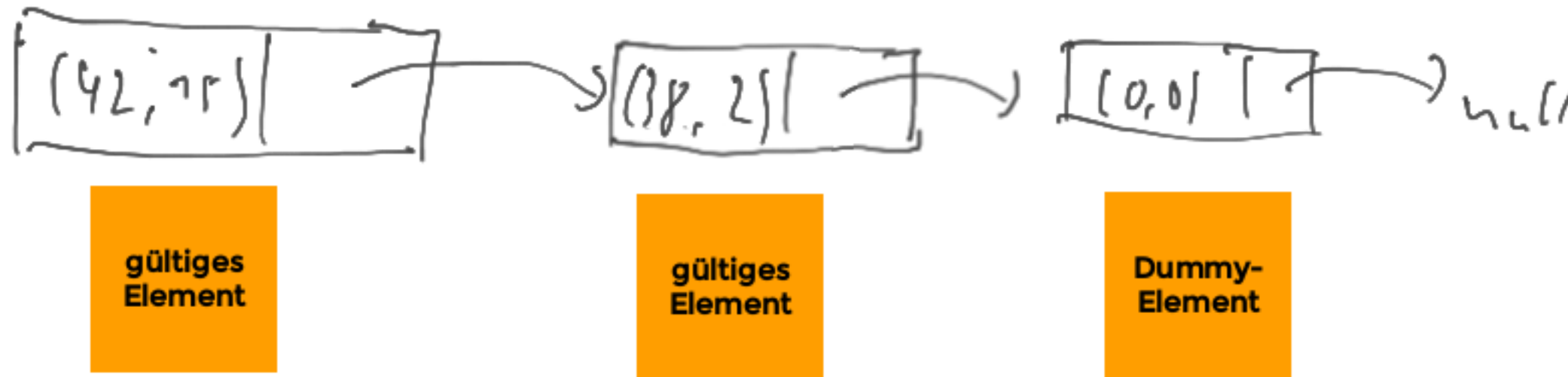
Dummy-
Element

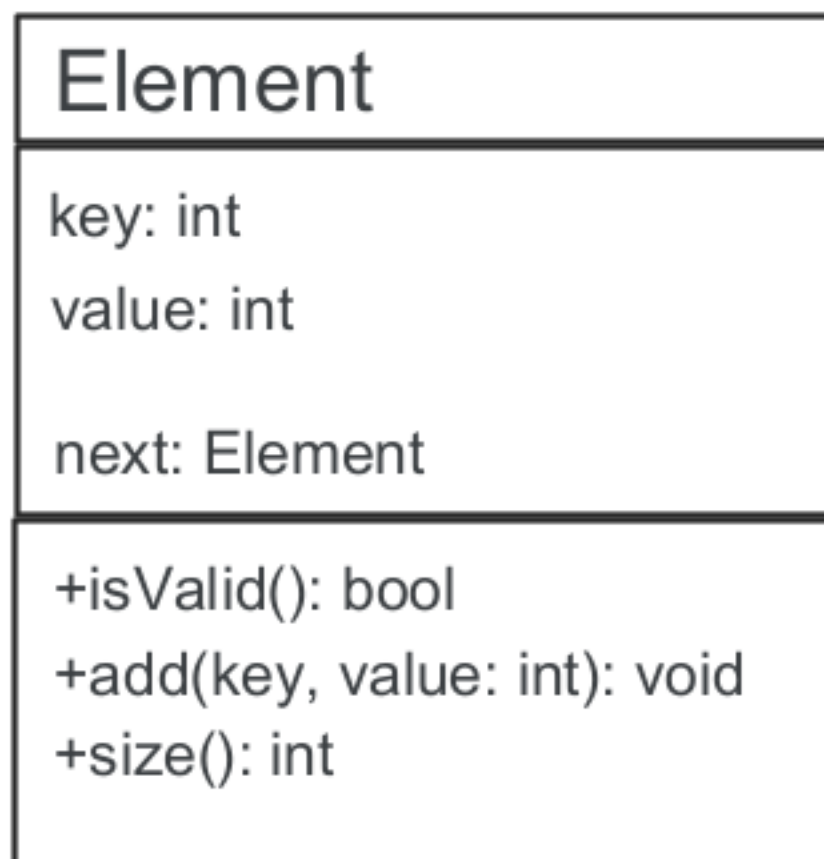
Nachfolger !=
null

Schrittweiser Aufbau

Anhängen eines neuen Elements

Zwei
Elemente





**Interface
für
Element**

-setData(key, value : int): void
-getEnd(): Element

Hilfsfunktionen

In
UML-Diagrammen
meist weggelassen.

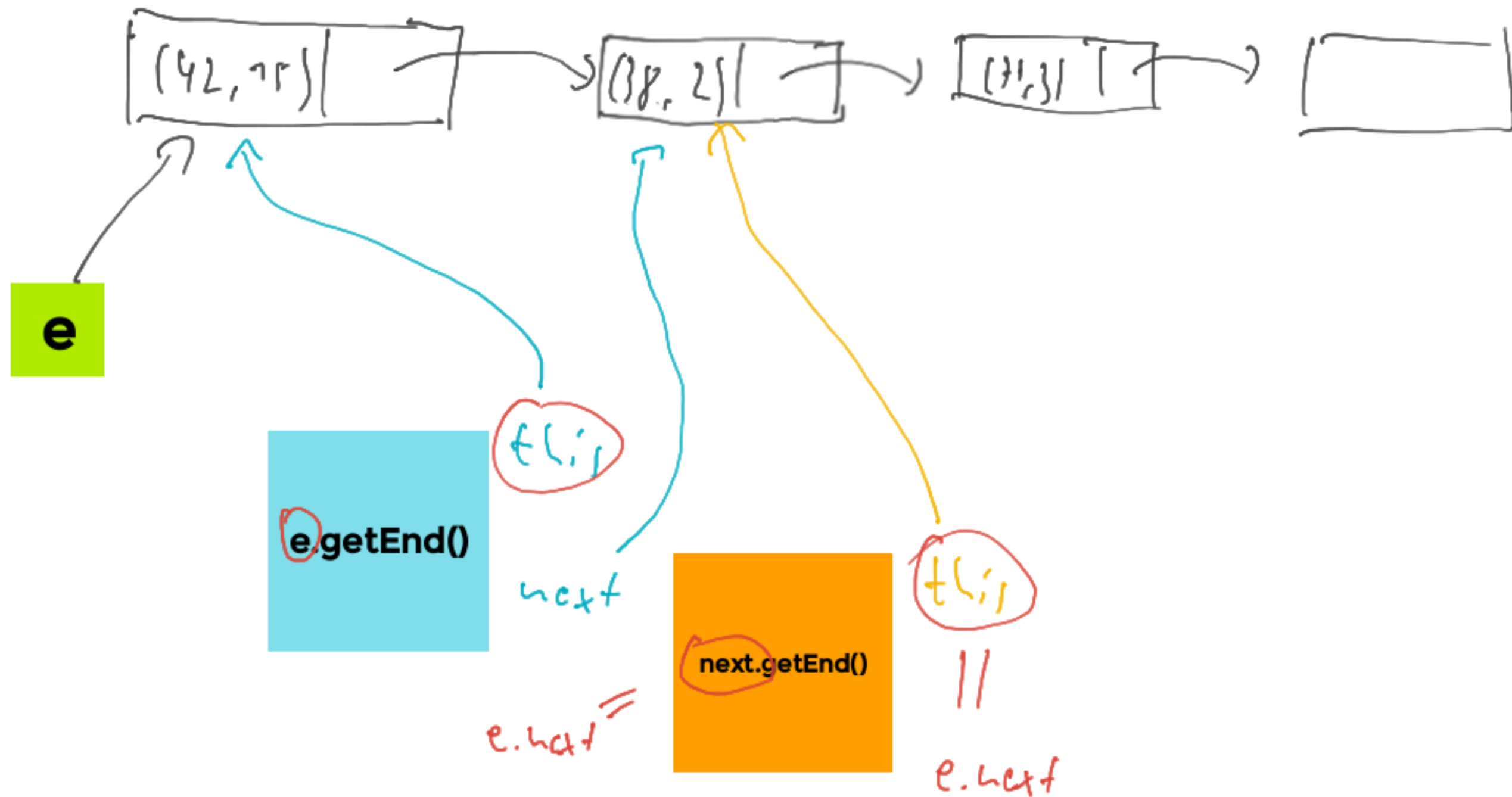
Klassendiagramm: Übersicht über die abstrakten Datentypen, die an einer Liste beteiligt sind.

Element
+isValid(): bool +add(key, value: int): void +size(): int

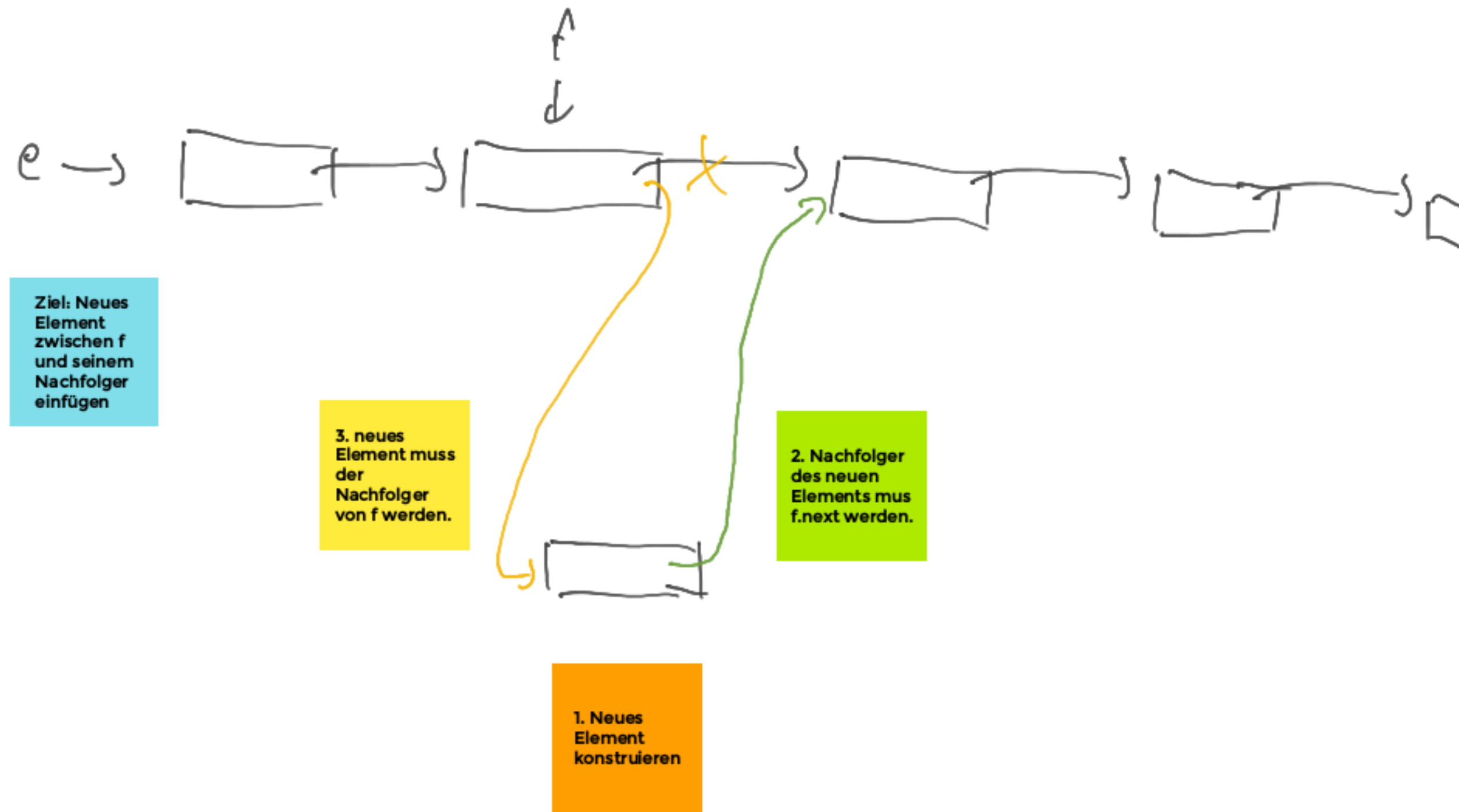
List
+add(key, value: int): void +size(): int

Schrittweiser Aufbau

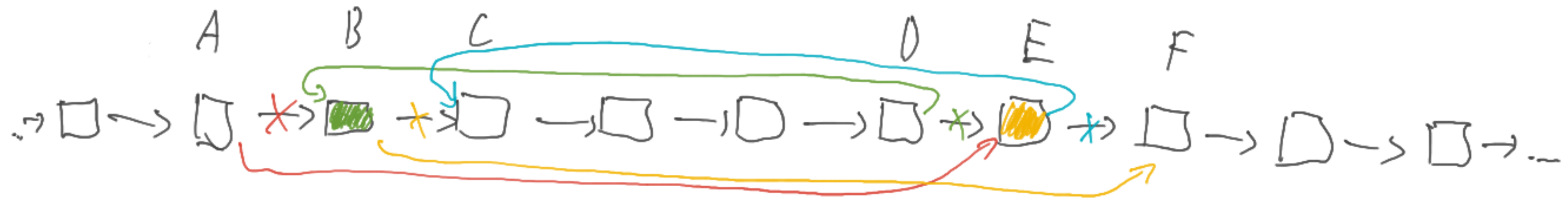
Finden des Endes



Einfügen von Elementen



Vertauschen von Elementen



A.next = E —

B.next = F —

D.next = B —

E.next = C —

Rekursive Natur von Listen

Eine
Liste
ist ...

... leer

oder

... ein Element gefolgt
von einer Liste

