

4 Rekursive Methoden einer Liste

Umfrage zum Fernsehverhalten (S.27)

Wie viele sind in der Liste?

rekursive Methode der Klasse Knoten:

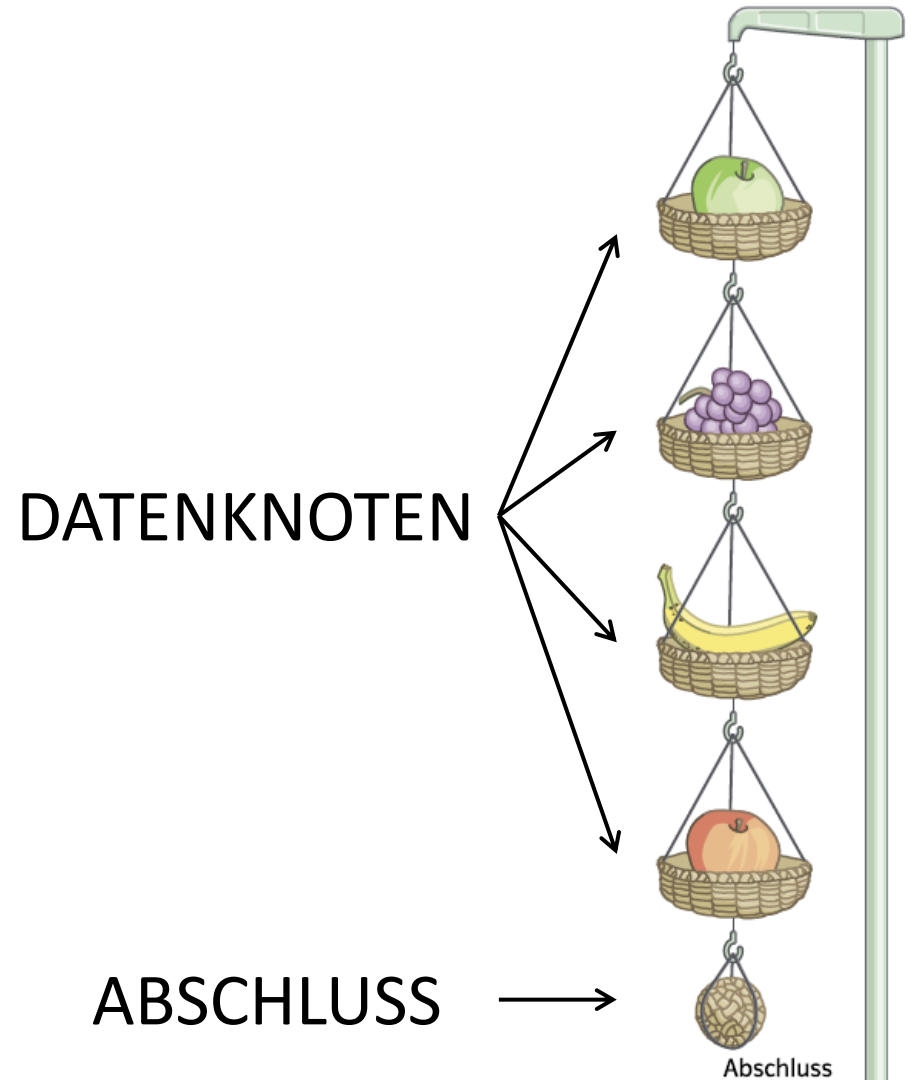
- falls ich der letzte bin, gebe ich 1 zurück.
- falls ich nicht der letzte bin, gebe ich die Frage an meinen Nachfolger weiter, warte auf seine Antwort, zähle 1 dazu und gebe das zurück.

Wie lange siehst du täglich fern (in Stunden)?

Alle Knoten bis auf den letzten verhalten sich gleich.

Idee:

Der letzte ist eine andere
Art von Knoten (Abschluss),
der keinen Dateninhalt hat.



Wie viele sind in der Liste?

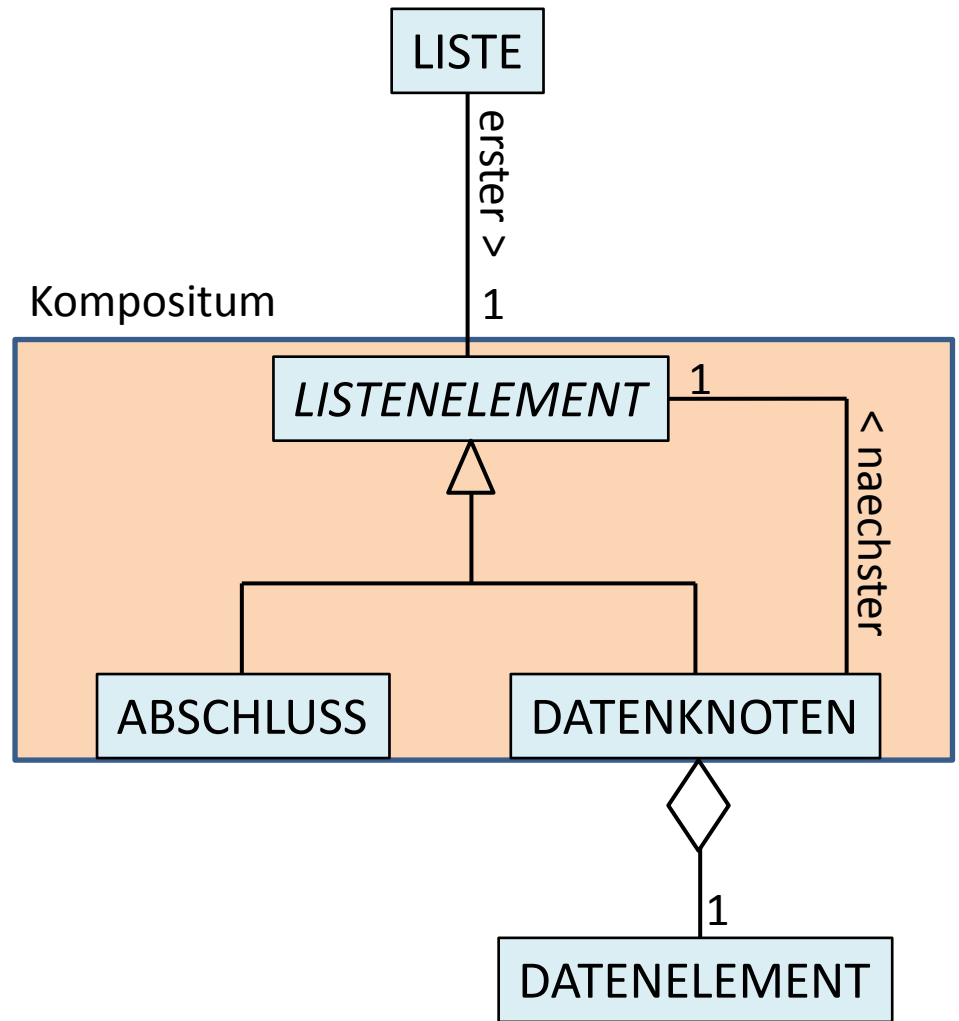
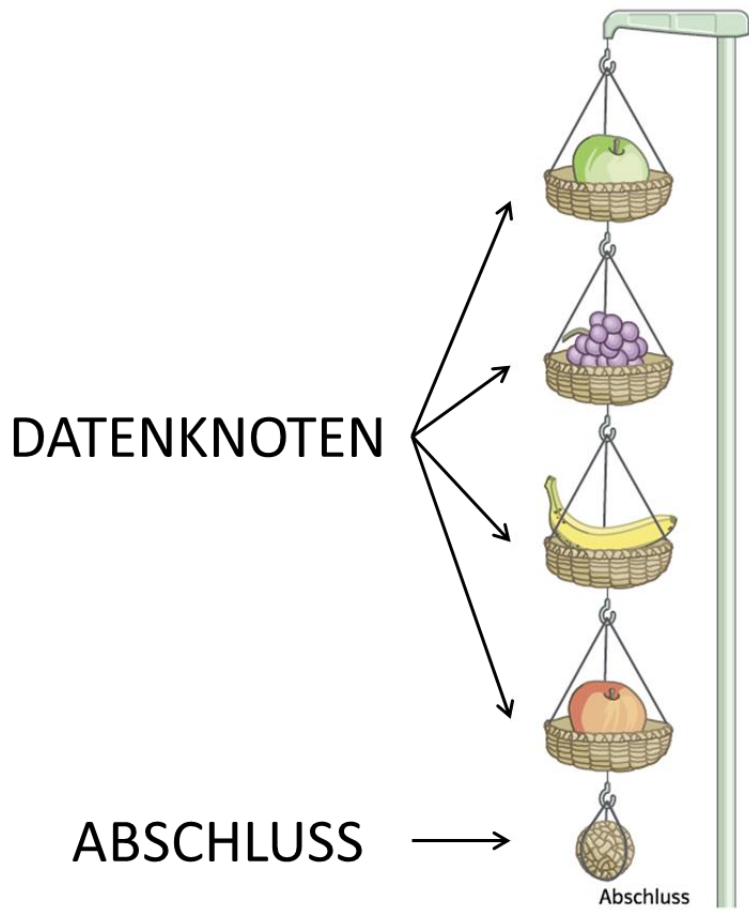
DATENKNOTEN:

Ich gebe die Frage an
meinen Nachfolger weiter,
warte auf seine Antwort,
zähle 1 dazu und gebe das
zurück.

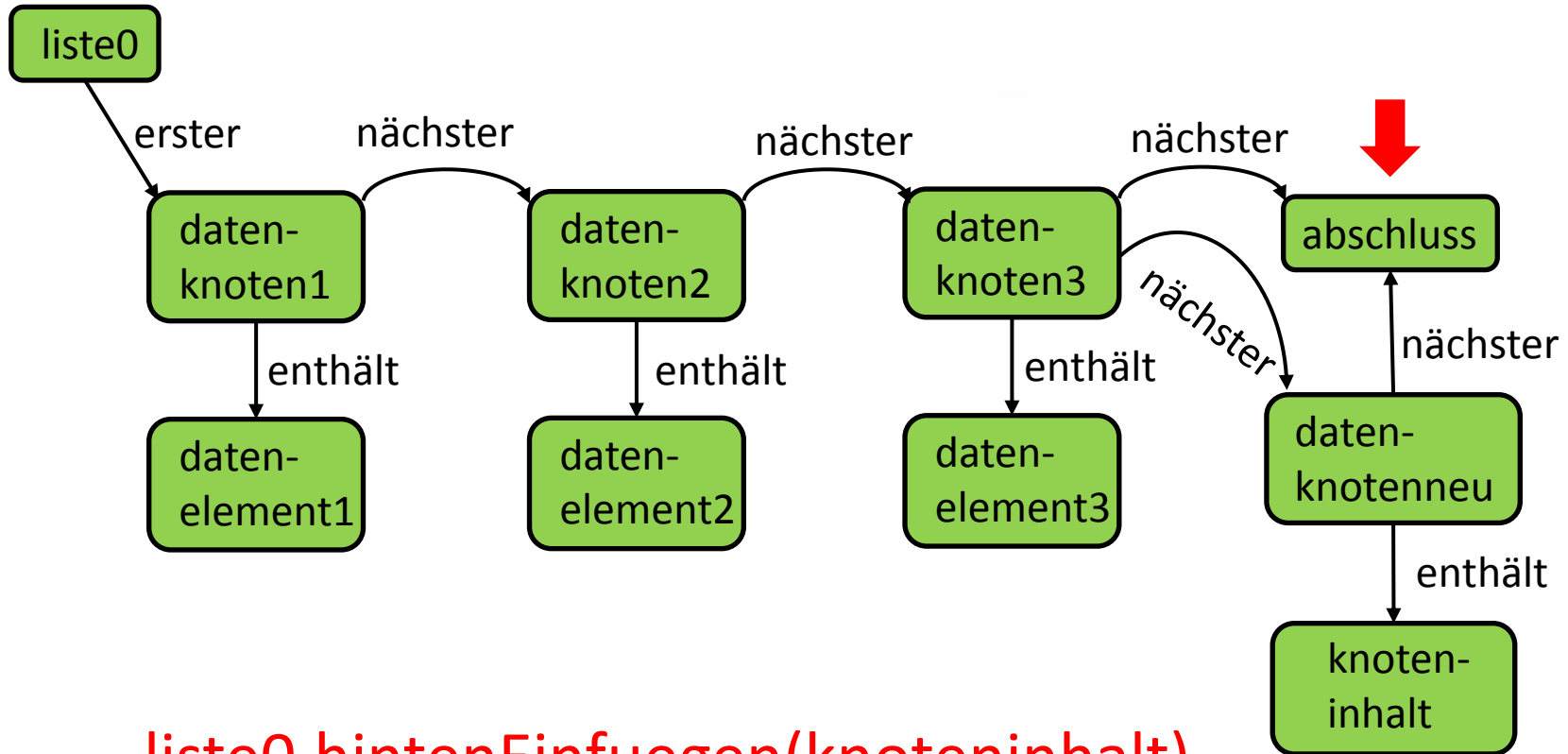
ABSCHLUSS:

Ich gebe 1 zurück.

Wegen der unterschiedlichen Methodendefinitionen
ist nun keine Fallunterscheidung mehr nötig!



hintenEinfuegen(Datenelement knoteninhalt)



liste0.hintenEinfuegen(knoteninhalt)

erster.hintenEinfuegen(knoteninhalt)

naechster.hintenEinfuegen(knoteninhalt)

Die Datenknoten reichen den neuen Inhalt nur weiter!

Die Hauptarbeit hat der Abschluss:

- Neuen Datenknoten erschaffen mit dem entsprechenden Inhalt und dem Abschluss (d.h. sich selbst, **this**) als Nachfolger.
- Dem ehemals letzten Datenknoten mitteilen, dass sein Nächster der neue Datenknoten ist.
Dazu gibt der Abschluss als Rückgabewert den neuen Datenknoten zurück.

Signatur:

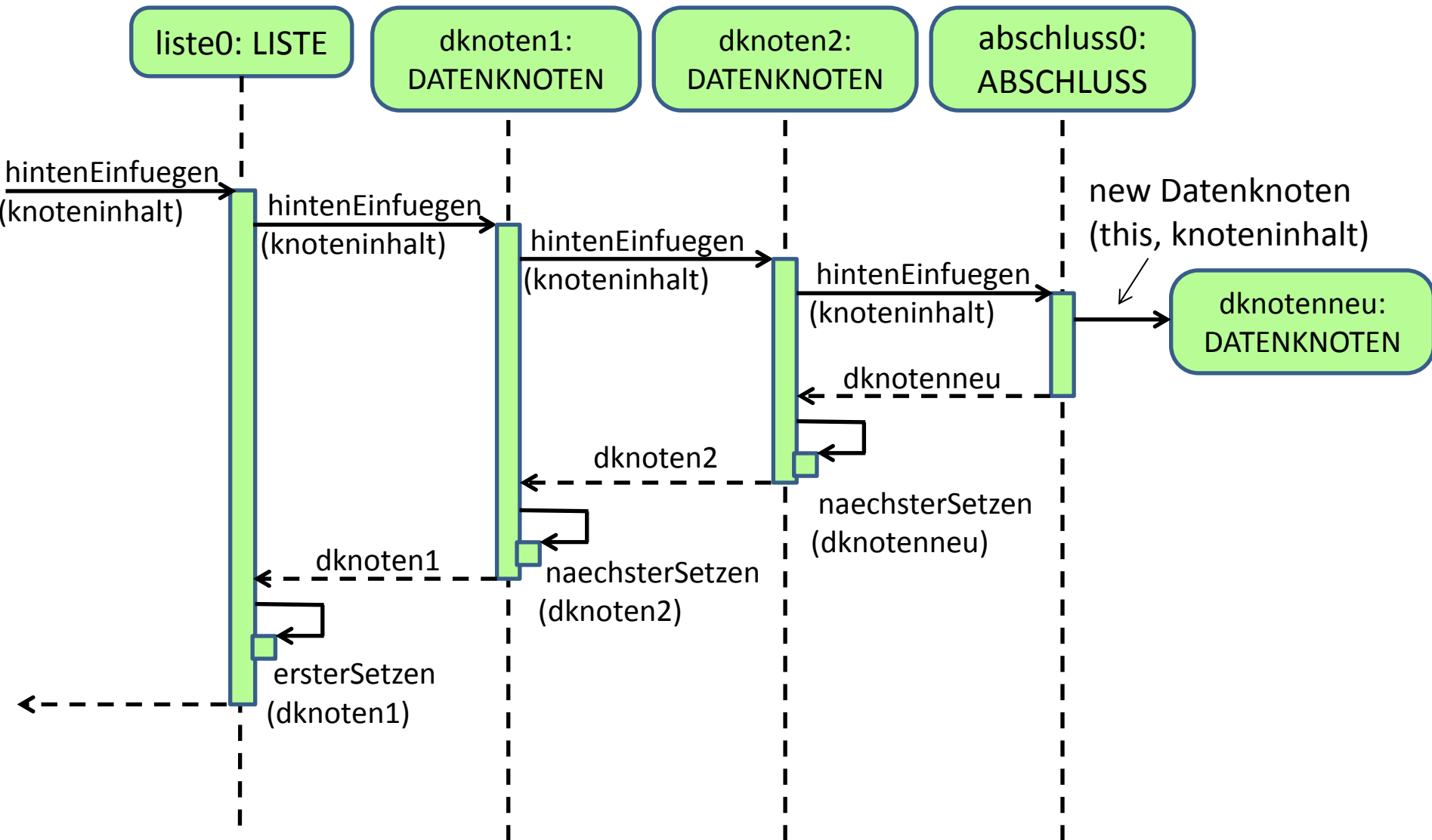
```
public Datenknoten hintenEinfuegen(Datenelement knoteninhalt)
```

Die Datenknoten erwarten also als Rückgabewert eine Referenz auf ihren nächsten.

Was geben alle Datenknoten an ihren Vorgängerknoten zurück?

Sich selbst: **this**

Sequenzdiagramm zu hintenEinfuegen(knoteninhalt)



Implementierung unter Verwendung des Kompositum:

<abstract> Listenelement
public abstract Datenknoten hintenEinfuegen(Datenelement knoteninhalt)

Datenknoten
private Datenelement inhalt private Listenelement naechster
public Datenknoten(Listenelement n, Datenelement i) public Datenknoten hintenEinfuegen(Datenelement knoteninhalt)

Abschluss
public Datenknoten hintenEinfuegen(Datenelement knoteninhalt)

Datenelement	Liste
bezeichner	private Listenelement erster
public Datenelement(String bez) public String datenGeben()	public Liste() public void hintenEinfuegen(Datenelement knoteninhalt)