



AU--
BA

17

GEOL--
GY

30

OT--
PE

BE--
BIO

HISTORY OF T

BIR--
BLY

19



Karteikarten -> Seiten



Intuition für B-Bäume

67	72	84	105
T	K	V	D

S
W

Eine weitere Seite...

67	72	84	105
T	K	V	D

34	56	83	87
S	Z	O	B

Umverteilen...

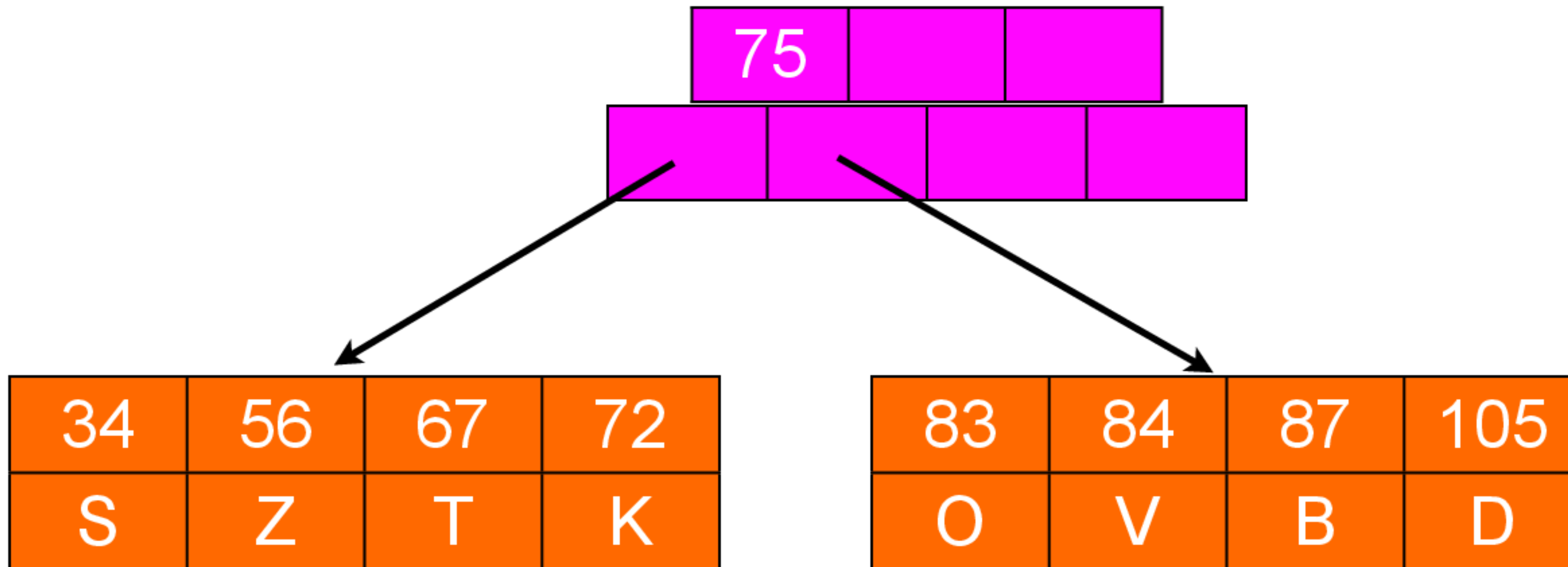
42

34	56	67	72
S	Z	T	K

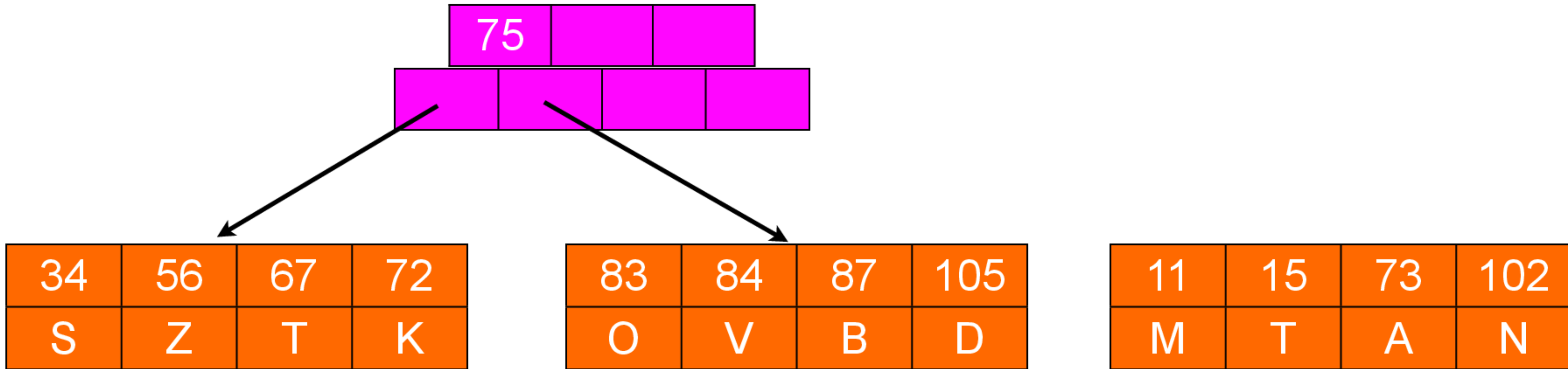
83	84	87	105
O	V	B	D

Deskriptor erzeugen...

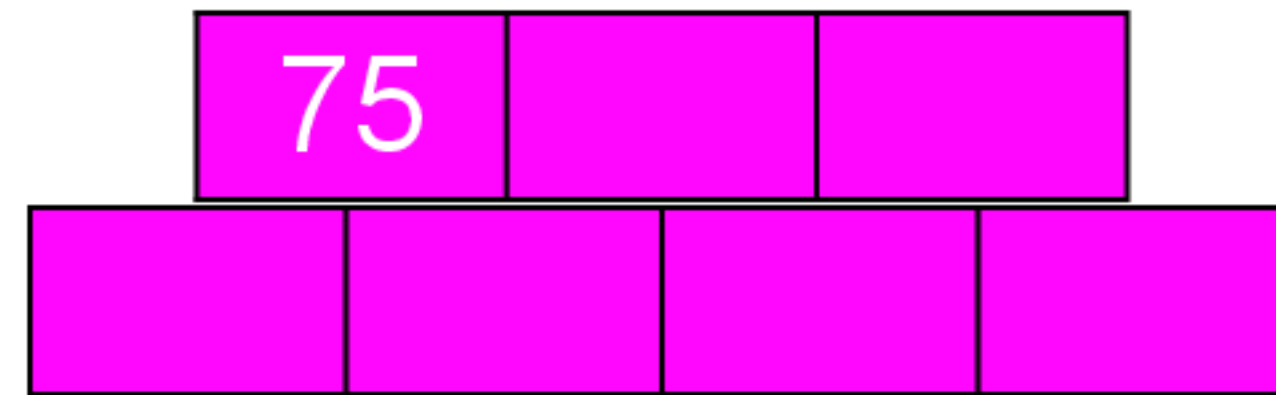
42



Eine dritte Seite...



Umverteilen...



11	15	34	56
M	T	S	Z

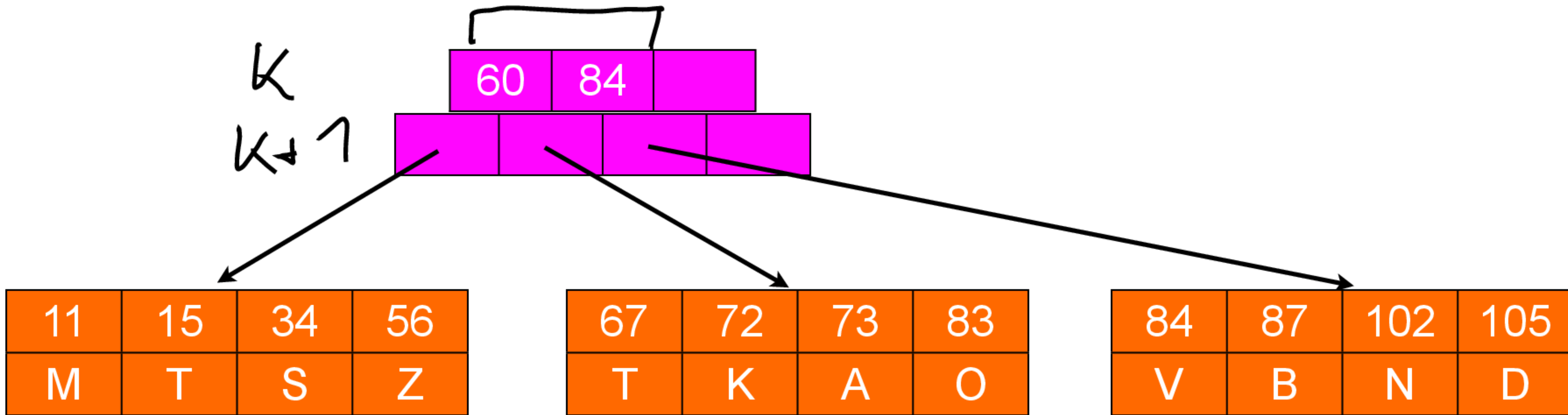
67	72	73	83
T	K	A	O

84	87	102	105
V	B	N	D

Deskriptor reparieren...

84

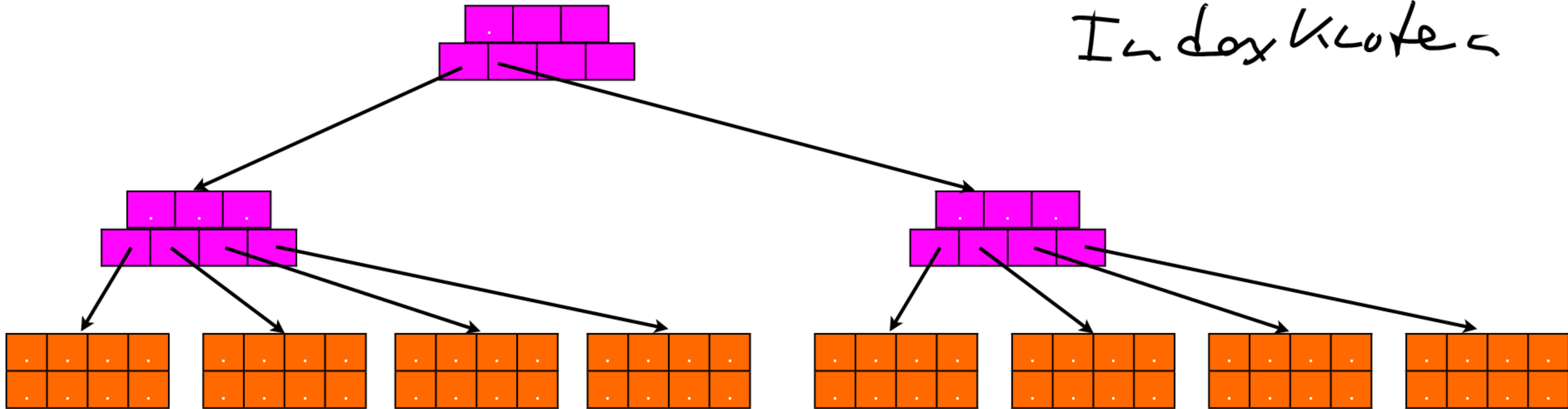
[60;84[



Rekursiv weiter...

Wurzelknoten

Indexknoten



Blattknoten

B-Tree Eigenschaften

Pfad von Wurzel zu Blatt: immer gleich lang

Indexknoten:

hat $k/2$ bis k Schlüssel

=> $k+1$ Kinder maximal

Blattknoten:

hat $k^*/2$ bis k^* Wert/Schlüssel-Einträge

min-Schranke $k/2$ und $k^*/2$ gelten nicht für Wurzel

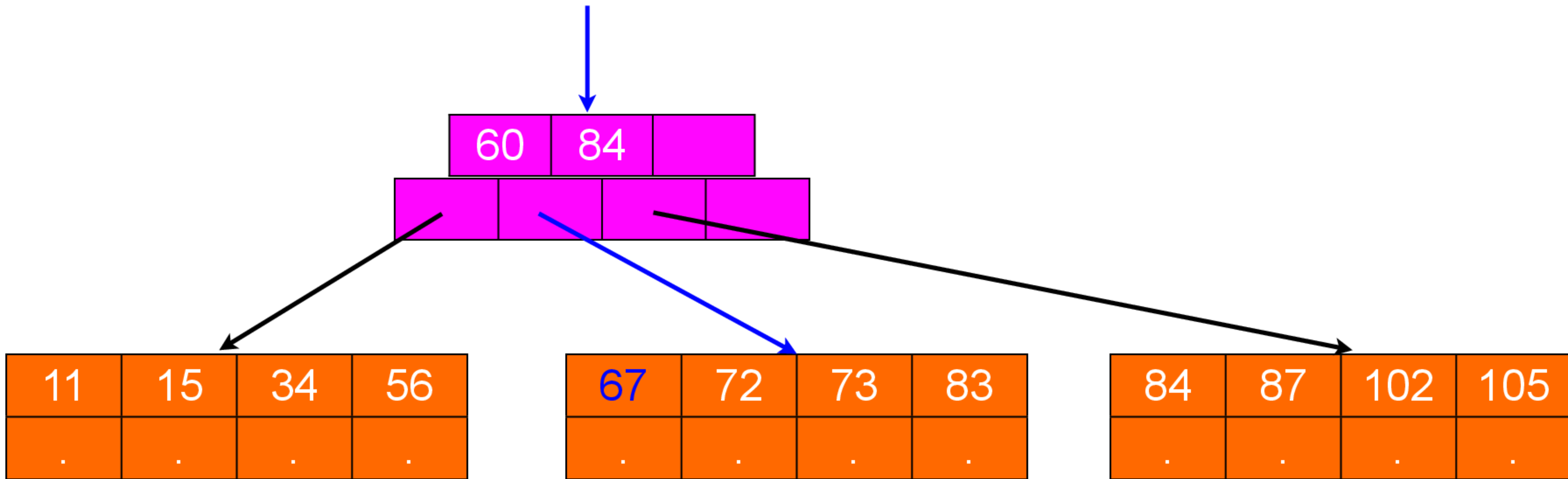
B-Tree Eigenschaften

keine Werte in Indexknoten

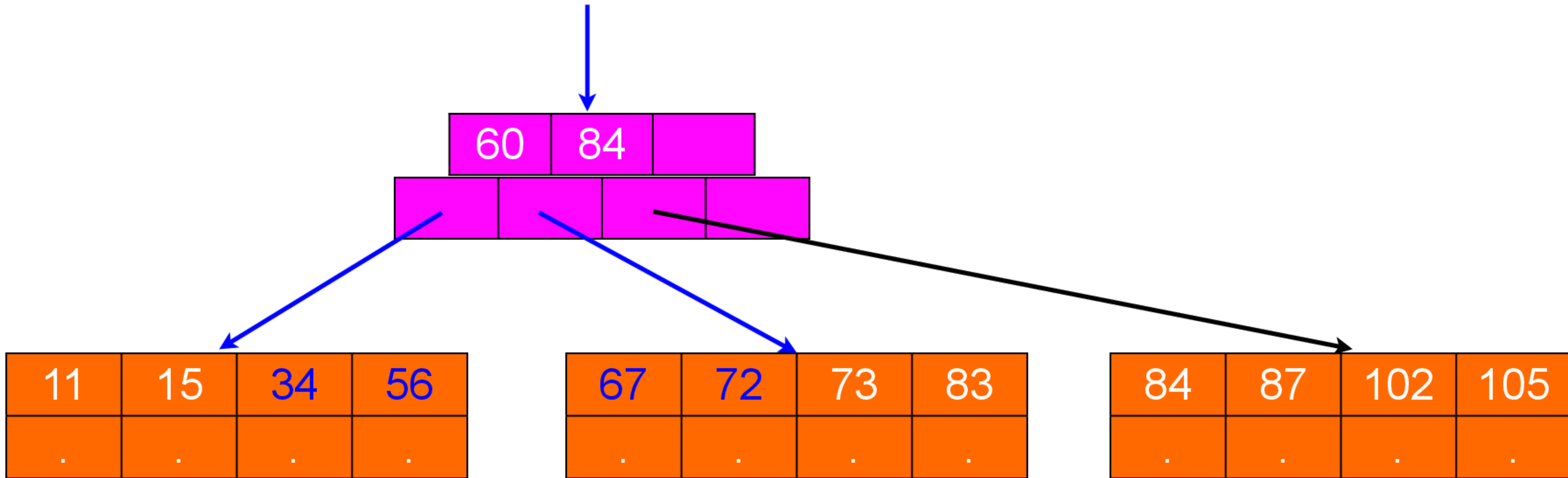
alle Daten in den Blattknoten

k und k^* implizit definiert durch Datentypen

find_key(67)

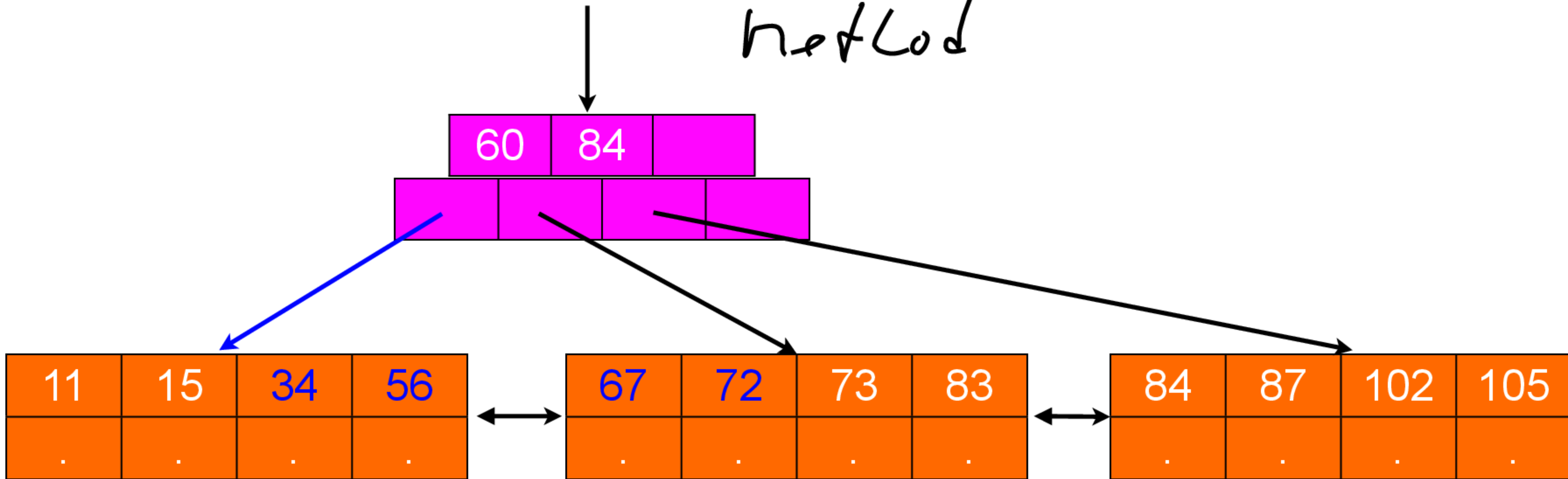


find_range([34;72])



ISAM: find_range([34;72])

Index
Sequential
Access
Method

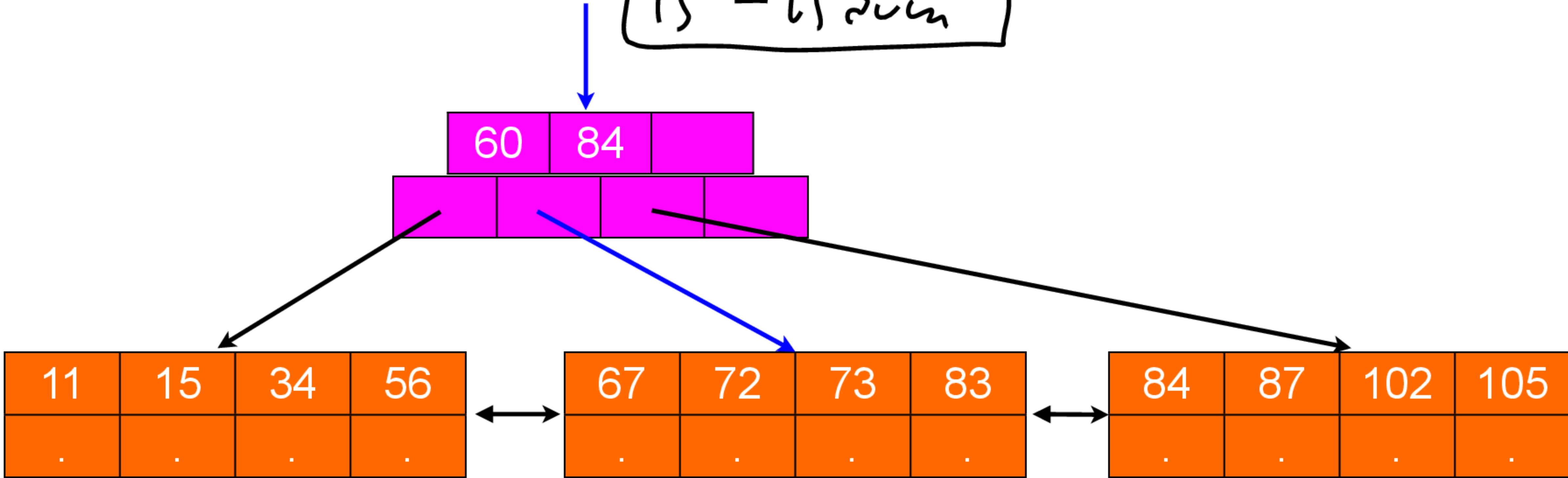


insert(74)

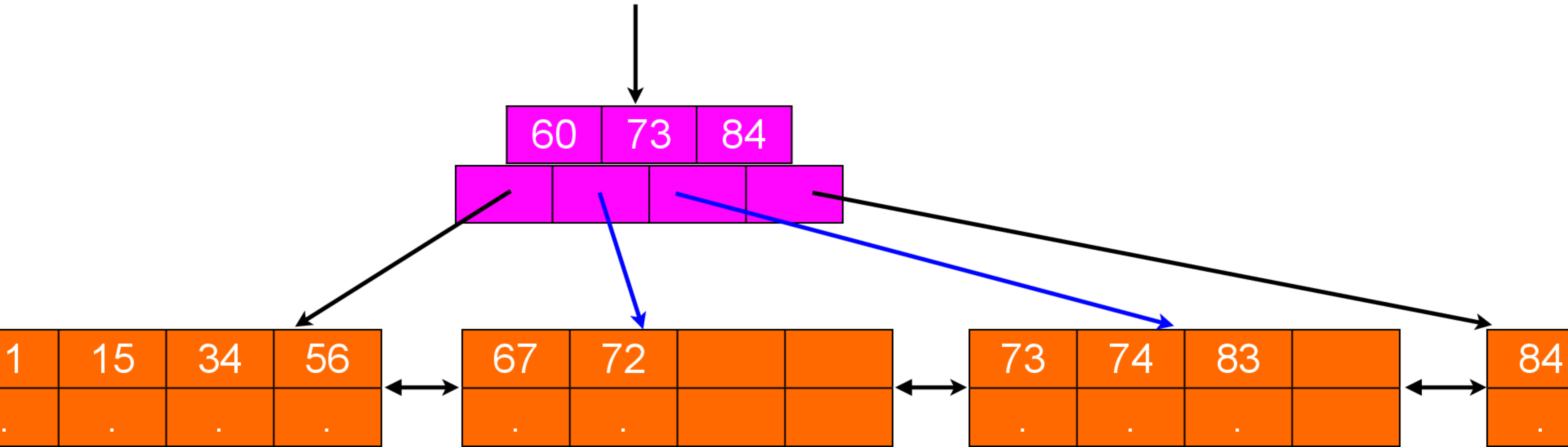
B - B sum

B* - B sum

B⁺ - B sum

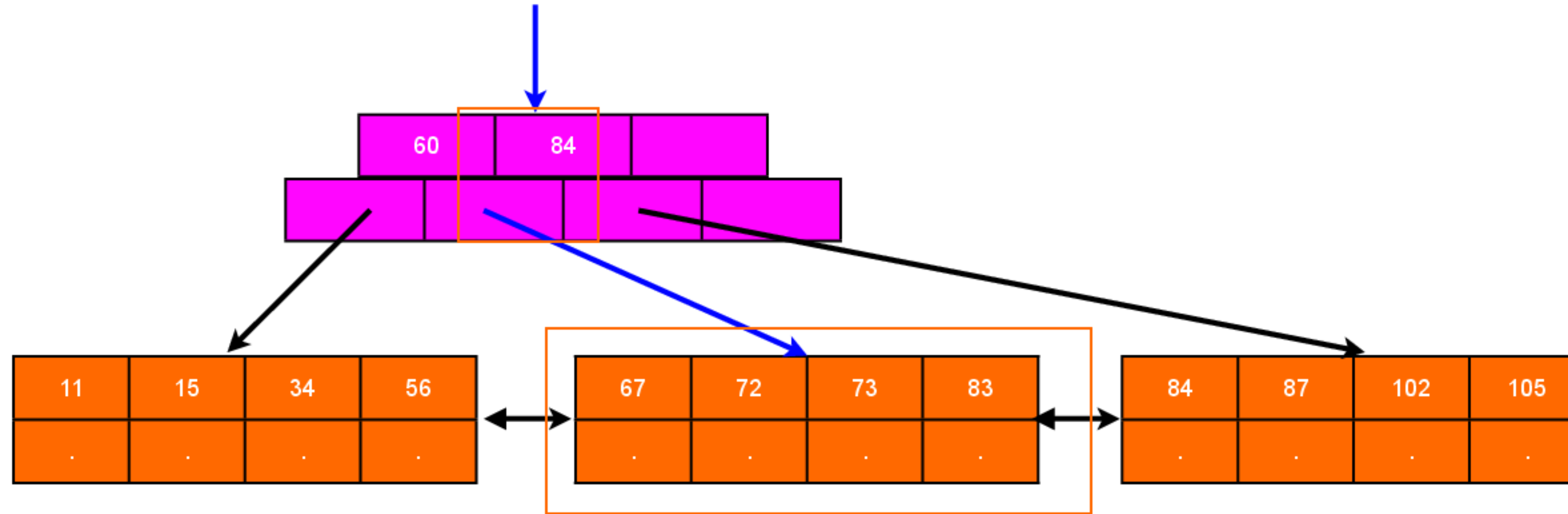


insert(74)

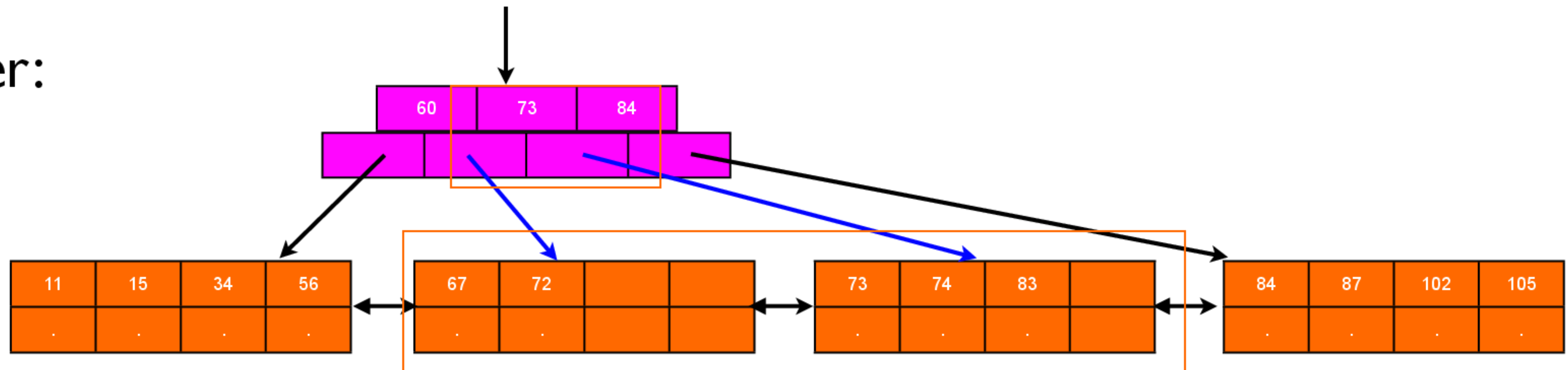


split()

vorher:



nachher:



BTree.insert(42, <value>)

Indexknoten.insert(42, <value>):

```
1. AbstractNode subtree = choose_subtree(42);
2. (newNode, pivot) = subtree.insert(42, <value>);
3. If newNode != null:
    insertEntry(newNode, pivot);
    If entries > k:
        return split();
4. return (null, null);
```

Blattknoten.insert (42, <value>):

```
1. insertEntry(42, <value>)
2. If entries > k*:
    return split();
Else:
    return (null, null);
```



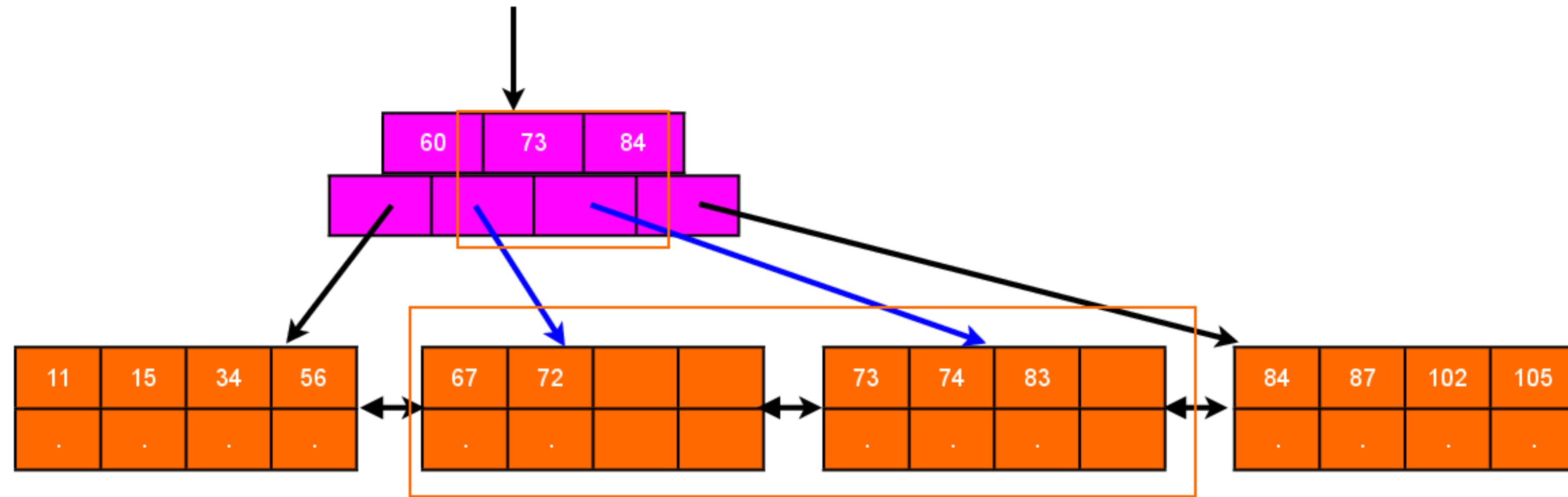
```
//Teilbaum auswählen
//insert an Teilbaum weiterleiten
//Gab es einen Split?
//Dann: neuen Eintrag hier einfügen
//Läuft dieser Knoten über?
//Split zum Vaterknoten hochgeben
//KeinSplit zum Vaterknoten hochgeben
```

```
//Neuen Eintrag hier einfügen
//Läuft dieser Blattknoten über?
//Split zum Vaterknoten hochgeben

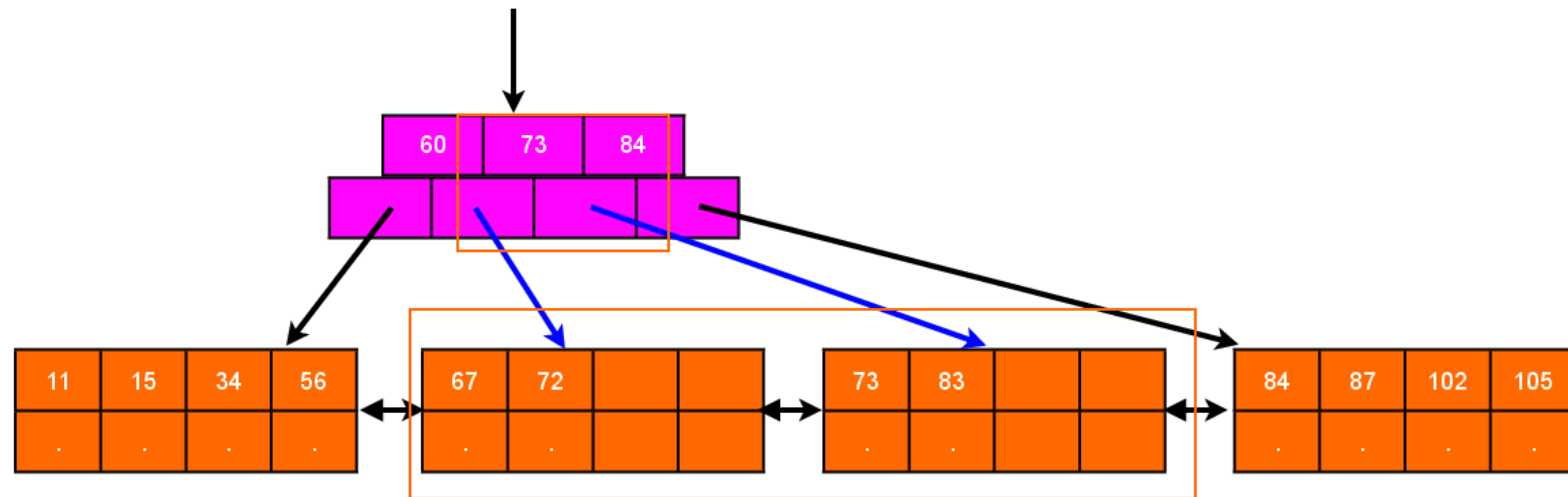
//KeinSplit zum Vaterknoten hochgeben
```

delete(74)

vorher:

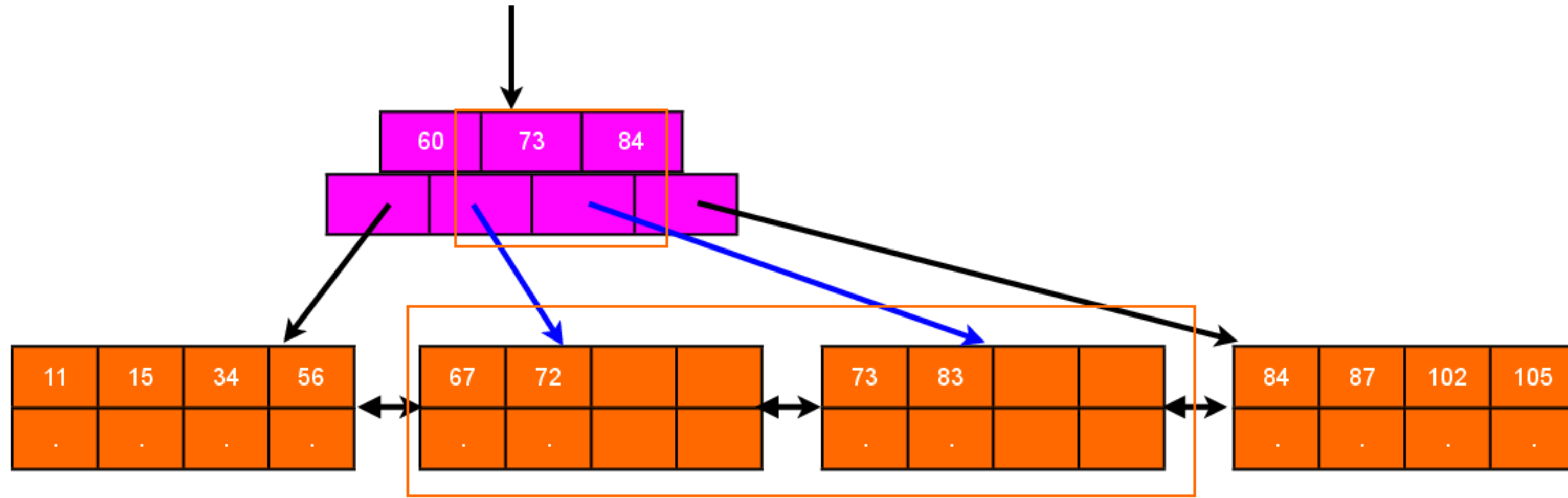


nachher:

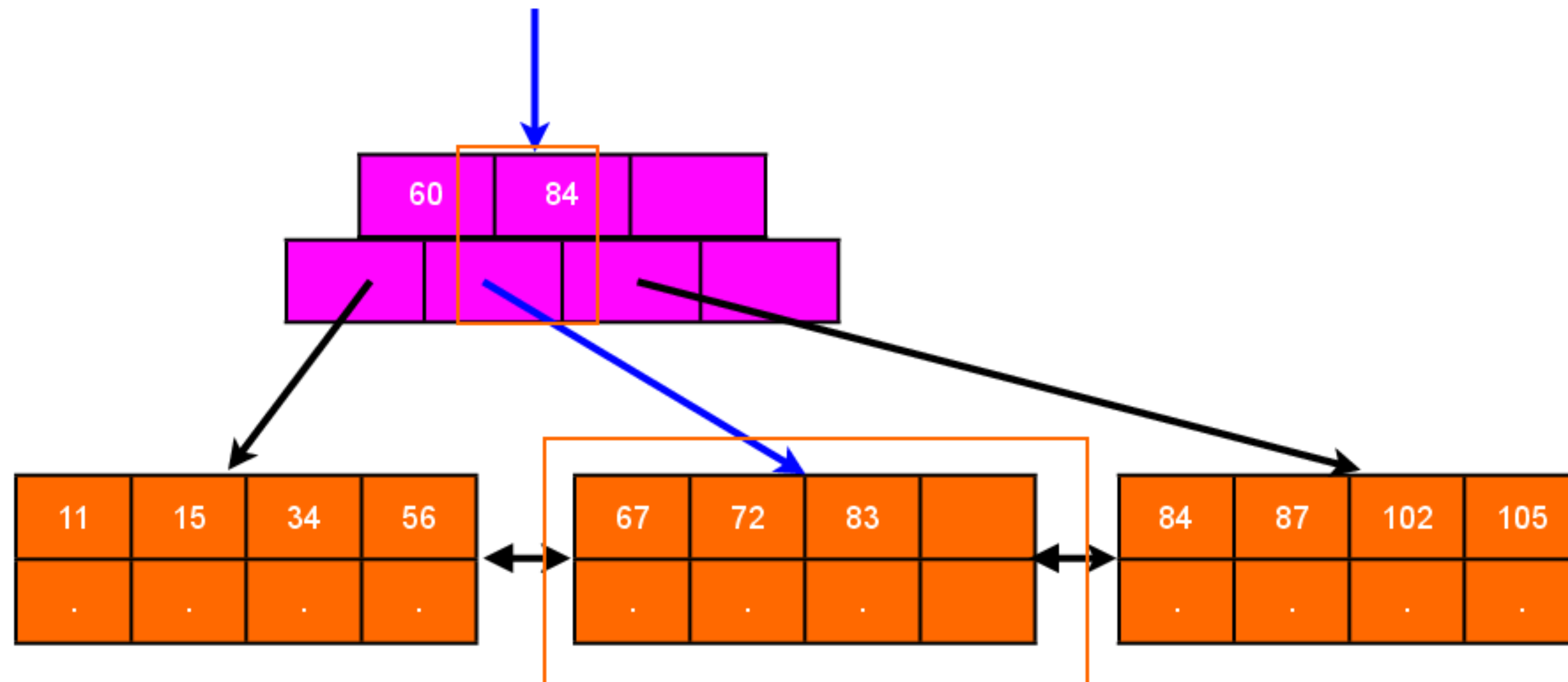


delete(73) -> merge()

vorher:



nachher:



delete() und merge()

delete():

=inverse Operation zu insert()

erst: find_key()

dann: Eintrag aus Blattknoten löschen

merge():

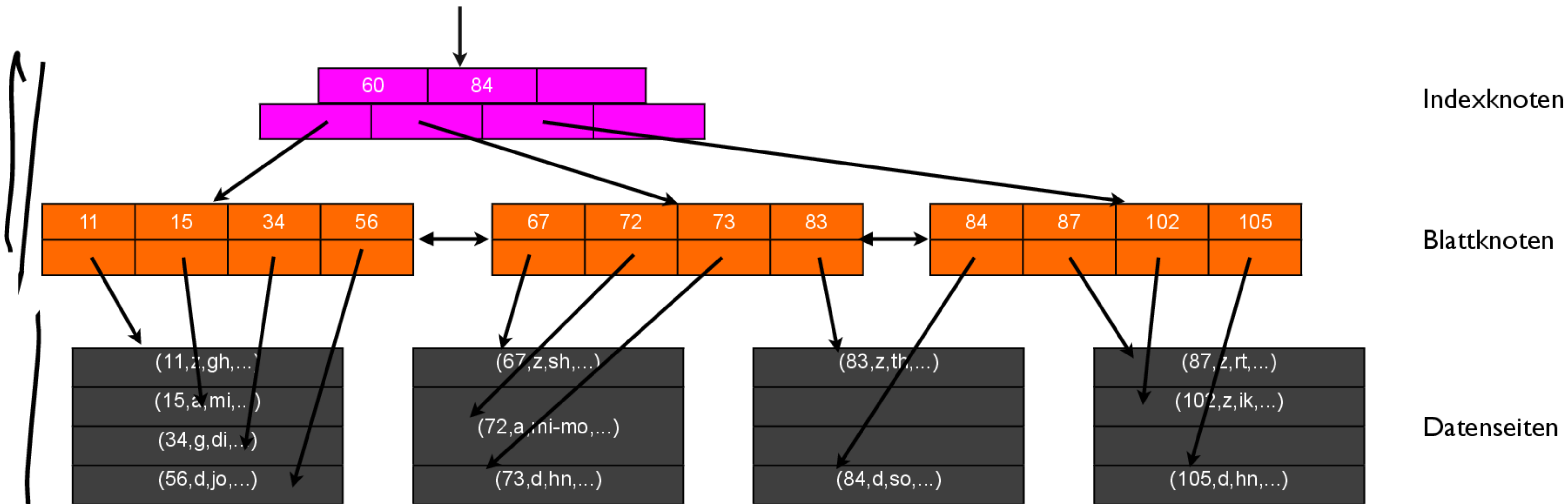
=inverse Operation zu split()

erst: zwei Knoten zusammenführen

dann: Eintrag vom Vaterindexknoten löschen

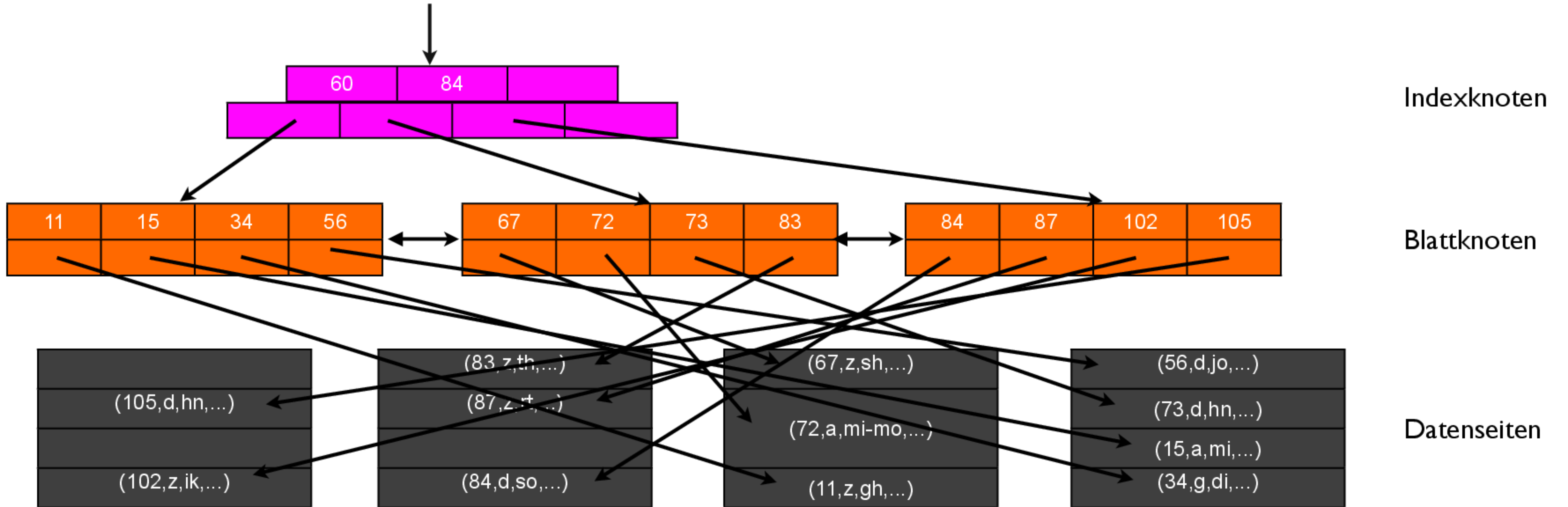
Clustered Index

[34; 87]



Unclustered Index

72



Copyrights and Credits

© **iStock.com:**

ftwitty; Funwithfood