

Algorithmen und Datenstrukturen (Th. Ottmann und P. Widmayer)

Folien: Splay-Bäume
Autor: Sven Schuierer

Institut für Informatik
Georges-Köhler-Allee
Albert-Ludwigs-Universität Freiburg

1 Splay-Bäume

Selbstanordnungsstrategien für:

- lineare Listen: Move-to-front, Transposition
- Bäume: Move-to-Root (durch wiederholtes Rotieren)
Problem: Kann für beliebig lange Zugriffsfolgen Kosten von $\Theta(n)$ Schritten pro Operation erfordern.

Splaybäume sind binäre Suchbäume, die sich durch eine Variante der Move-to-root Strategie selbst anordnen.

2 Die Splay-Operation

T.Splay(x):

1. Suche nach x in T

if x ist in T

then Sei p der Knoten mit Schlüssel x

else Sei p der Vater des Blattes, bei dem die Suche endet

2. Wiederhole Operationen

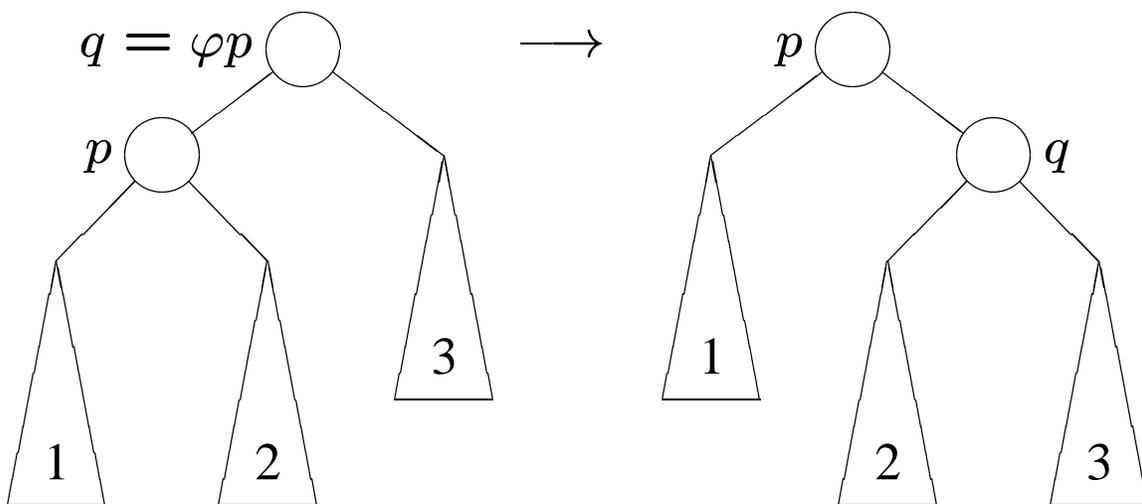
zig, zig-zig, zig-zag

bis p zur Wurzel geworden ist

3 Die Splay-Operation

Fall 1: p hat Vater φp und φp ist die Wurzel.

Operation **zig**: Rotation nach links oder rechts, die p zur Wurzel macht.

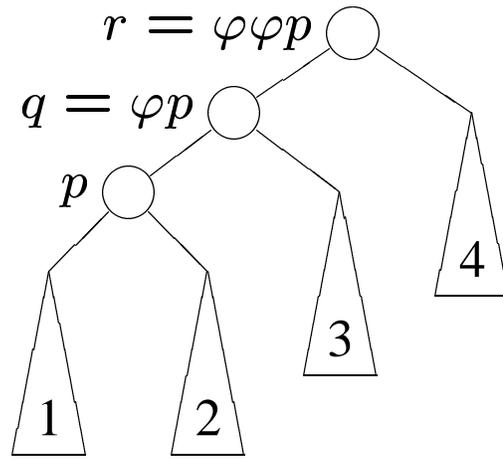


4 Die Splay-Operation

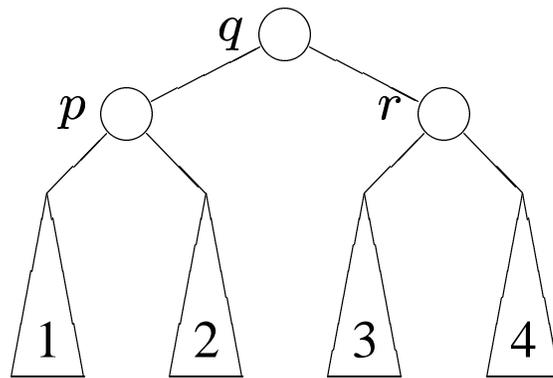
Fall 2:

p hat Vater φp und Großvater $\varphi\varphi p$ und p und φp sind beides **rechte** oder beides **linke** Söhne.

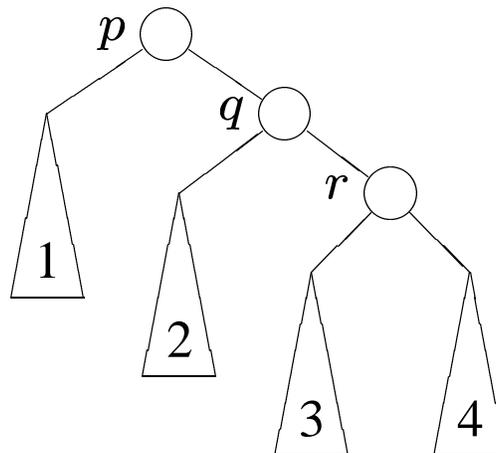
Operation **zig-zig**: zwei aufeinanderfolgende Rotationen in dieselbe Richtung, die p zwei Niveaus hinaufbewegen.



→
 Rotation
 nach rechts
 bei r



→
 Rotation
 nach rechts
 bei q

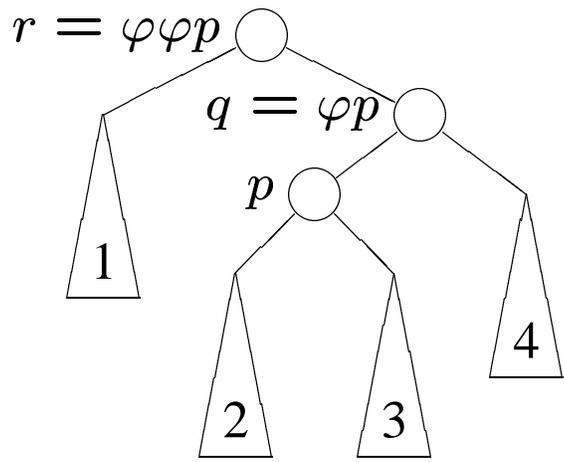


Die Splay-Operation

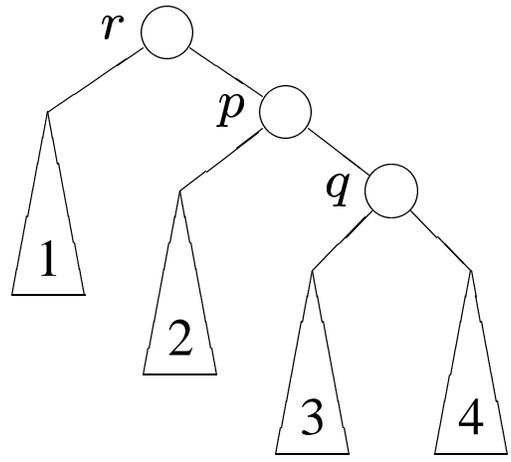
Fall 3:

p hat Vater φp und Großvater $\varphi\varphi p$ und einer der beiden Knoten p und φp ist linker und der andere rechter Sohn seines jeweiligen Vaters

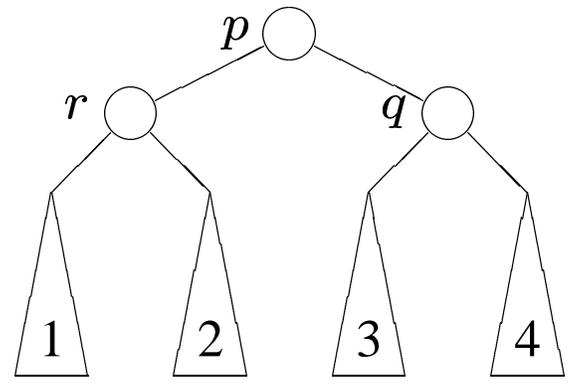
Operation **zig-zag**: zwei Rotationen in entgegengesetzte Richtungen, die p zwei Niveaus hinaufbewegen.



→
 Rotation
 nach rechts
 bei q



→
 Rotation
 nach links
 bei r



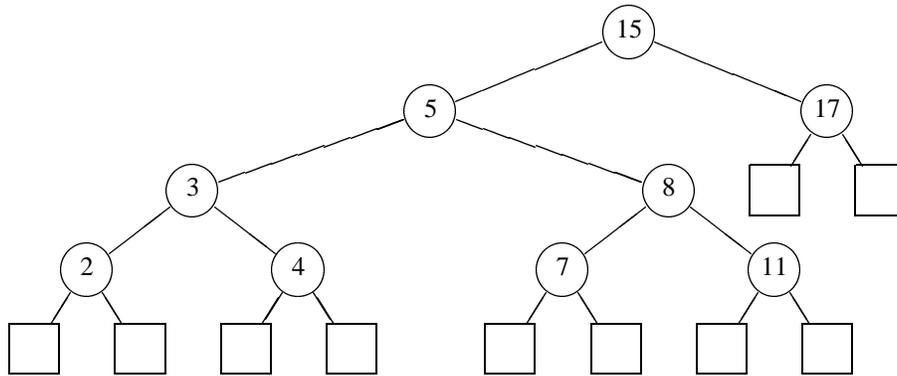
5 Die Splay-Operation

Wirkung von $T.\text{Splay}(x)$

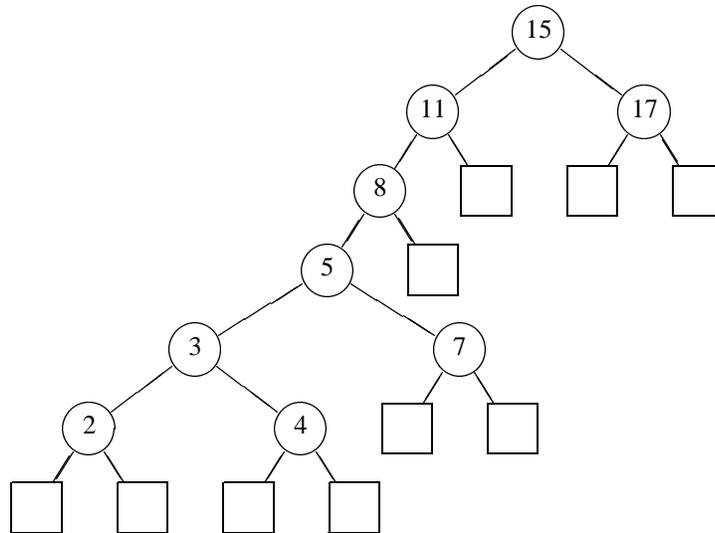
- **Fall 1:** x kommt in T vor
 x wird Schlüssel der Wurzel
- **Fall 2:** x kommt nicht in T vor
symmetrischer Vorgänger (oder Nachfolger) von x
wird Schlüssel der Wurzel

6 Beispiel

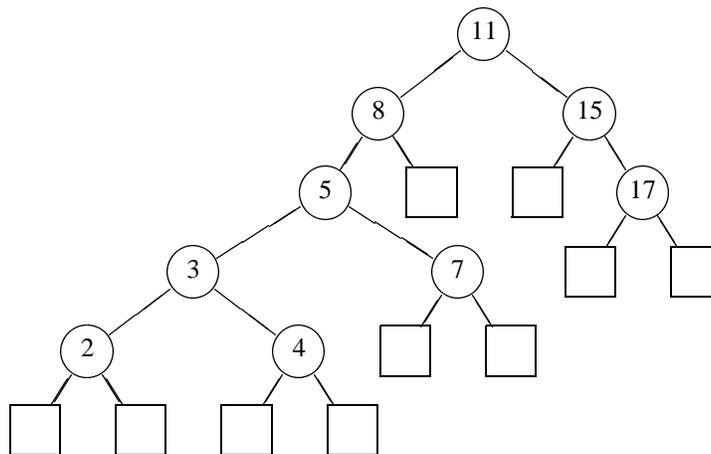
T.Splay(11)



→
zig-zig



→
zig



7 Wörterbuchoperationen

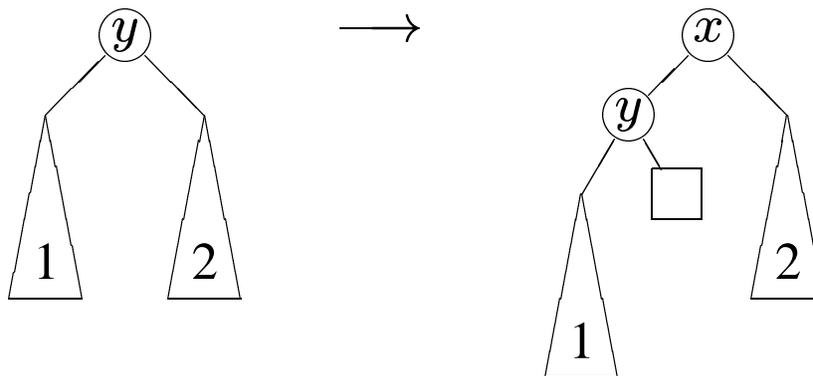
Suchen nach x in T :

1. T.Splay(x)
2. Vergleiche x mit Schlüssel der Wurzel

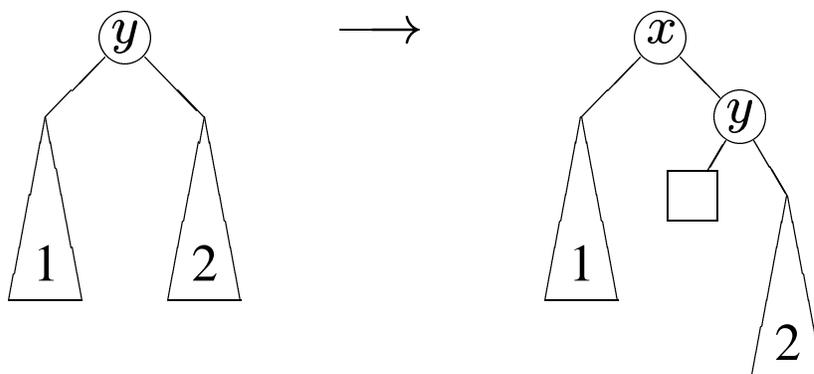
Einfügen von x in T :

T.Splay(x)

y ist symmetrischer Vorgänger:



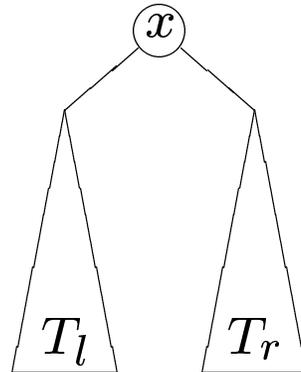
y ist symmetrischer Nachfolger:



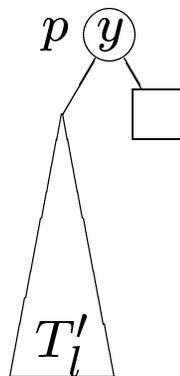
Wörterbuchoperationen

Entfernen von x aus T :

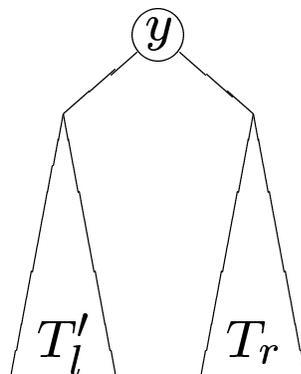
1. $T.\text{Splay}(x)$



2. $T_l.\text{Splay}(x)$



3. $p.\text{right} = T_r$



8 Eigenschaften von Splay-Bäumen

Satz

Führt man für einen beliebigen binären Suchbaum mit n Schlüsseln m -mal die Operation **Suchen** aus, so ist die dafür insgesamt benötigte Zeit von der Größenordnung $O((n + m) \log n)$.

Satz

Führt man für einen beliebigen binären Suchbaum mit n Schlüsseln insgesamt m -mal die Operation **Suchen** aus, so daß dabei auf Schlüssel i q_i -mal zugegriffen wird mit $q_i \geq 1$, so ist die dafür insgesamt benötigte Zeit von der Größenordnung

$$O\left(m + \sum_{i=1}^n q_i \log\left(\frac{m}{q_i}\right)\right).$$