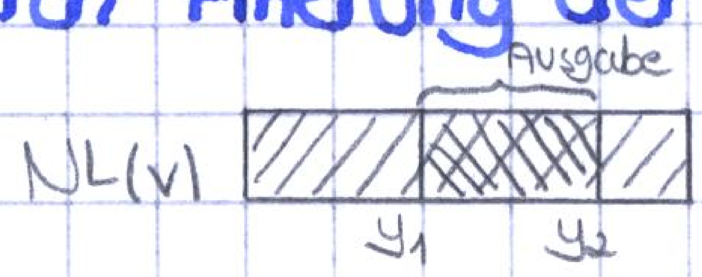
- **Ausgabe:**
  $\{p=(p_x,p_y)\in S : x_1 \le p_x \le x_2 ,\ y_1 \le p_y \le y_2\}$
  Dh alle Pkte $p\in S$ die im Rechteck $(x_1 y_1 ,\ x_2 y_2)$ liegen.

- **Datenstruktur:**
  Blattorientierter Suchbaum, wobei jeder Knoten v eine nach y-Koord sortierte Folge NL(v) von Pktn speichert → Knotenliste.
  - Platz: $O(n\cdot \log N + N)$
  - Bereichsabfrage: $O(\log N \cdot \log n + k) = O(\sum_{v\in C(x_1,x_2)} (\log n + k_v))$
  - Einfügen: $O(\log N \cdot \log n)$
  - Löschen: $O(\log N \cdot \underbrace{\log n}_{\text{Einfügen/Löschen in Liste NL}})$

- **2-dim Bereichsabfrage:**
  1.) • Alle Pkte im unendlichen Streifen zwischen x_1 und x_2
  Beh.: Die Pkte in allen Knotenlisten NL(v) für $v\in C(x_1,x_2)$ sind genau die Pkte im Streifen
  Beweis: Für bel. Knoten v enthält NL(v) alle Pkte mit x-Koord im Bereich der Blätter des Unterbaums von v.
  Außerdem bilden die Bereiche der Knoten $v\in C(x_1,x_2)$ eine Partition des Intervalls $[x_1,x_2]$
  2.) • Ausgabe aller Pkte mit y-Koord aus $[y_1,y_2]$ durch Filterung der Knotenlisten NL(v) $\forall v\in C(x_1,x_2)$.



5.2.3 d-dim

- **Datenstruktur:**
  Blattorientierter Suchbaum für die 1. Koordinate.
  Jeder Knoten v speichert NL(v) als (d-1)-dim Range-Tree.
  Jeder Pkt $p\in S$ wird in allen NL(v), für v auf Suchpfad nach 1. Koord. von p, gespeichert.

- **VS:** gilt nur zunächst
  Jede Dimension hat Koord $\{1,\dots,N\}$

- **Kosten:**
  - Platz: $O(\log N \cdot \text{Platz}_{d-1}(n)) = O(n\cdot \log^{d-1} N + N)$
  - Bereichsabfrage: $O(\sum_{v\in C(x_1,x_2)} (\text{Abfrage}_{d-1}(n) + k_v)) = O(\log^{d-1} N \cdot \log n + k)$
  - Einfügen: $O(\log N \cdot \text{Einfügen}_{d-1}(n)) = O(\log^{d-1} N \cdot \log n)$
  - Löschen: $O(\log N \cdot \text{Löschen}_{d-1}(n)) = O(\log^{d-1} N \cdot \log n)$

5.3 Bemerkungen zu Segment- und Rangebäumen

- ∃ auch voll dynamische Varianten dh ∄ Gitter $\{1,\dots,N\}$ sondern relle Koord.
  Problem: Rebalancierung nach Einfügen / Löschen

- **Kosten allgemein:**
  - Platz: $O(n\cdot \log^{d-1} n)$
  - Bereichsabfrage: $O(\log^d n + k)$
  - Einfügen / Löschen: $O(\log^d n)$

BRUNNEN
Papier und viel mehr.