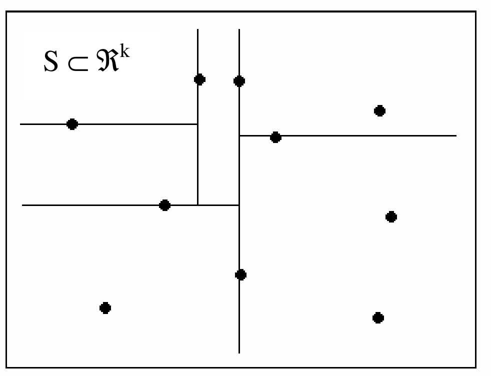
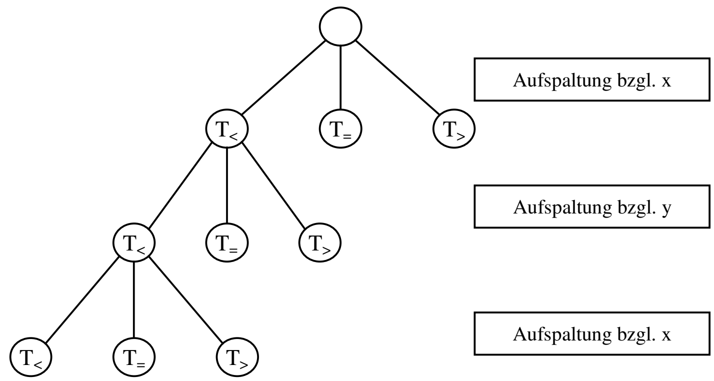
Thema: orthogonale Bereichsabfragen

Wiederholung: *kd-Bäume*





Sei S eine Teilmenge aus $\mathfrak{R}^k$ mit |S| = n. Ein *kd-Baum* realisiert eine Datenstruktur, mit der man in der Menge S effizient nach mehreren Schlüsseln (nach x- und y-Koordinate) suchen kann.
Ein *kd-Baum* ist folgendermaßen definiert:

- der Minimalfall ist k = n = 1: S besteht nur aus einem Element und der zugehörige *kd-Baum* aus einem einzelnen Blatt.
- für k>1 oder n>1 gilt:
  - die Wurzel des Baumes ist markiert durch die erste Koordinate, nach der alle Punkte in drei Teilbäume aufgeteilt werden in $T_<$, $T_=$ und $T_>$; in $T_<$ befinden sich die Elemente von S, die eine kleinere erste Koordinate haben als das Referenzelement, in $T_=$ die Elemente, die den gleichen Koordinatenwert haben und in $T_>$ diejenigen, die in der ersten Koordinate einen größeren Wert haben
  - diese Teilbäume werden dann wieder aufgeteilt, jedoch nach der zweiten Koordinate
  - usw.
- die Höhe des *kd-Baumes* beträgt O(log n+k)
- in den $T_=$ - Bäumen verringert sich die Dimension jeweils um 1
- um die Position der Punkte zu festzuhalten, werden in jedes Blatt, das am Ende des Baumes übrig bleibt, die Koordinaten der Punkte gespeichert

Satz:

a) Für eine gegebene Menge S, |S| = n, kann ein balancierter *kd-Baum* in O(n (log n+k)) Zeit mit Platzbedarf O(kn) konstruiert werden (i.a. ist k konstant → O(n log n) Verarbeitungszeit und Platzbedarf O(n)).
b) Für k = 2 kann eine orthogonale Bereichsabfrage mit Bereich U in O($\sqrt{n}$ + m) Zeit mit m = $|U \cap S|$ beantwortet werden.

<u>Algorithmus für a):</u>

- falls $|S| = 1 \Rightarrow S = \{p\}$ p

- andernfalls:

  - finde den Median der Koordinate von S mit der begonnen werden soll
  - beschrifte mit diesem Wert die Wurzel
  - teile S auf in $S_<$, $S_=$ und $S_>$ bezüglich dieser Koordinate
  - konstruiere rekursiv $T_<$, $T_=$ und $T_>$

<u>Gesamtlaufzeit:</u>

- pro konstruiertem Knoten v: $O(|S(v)|)$ zur Mediansuche und Aufteilung von S
  ($|S(v)|$ = Menge aller Blätter im Teilbaum mit Wurzel v)
- insgesamt $O(\sum_{\text{Knoten } v} |S(v)|) = O(\Sigma_m \Sigma_{\text{Tiefe}(v)=m} |S(v)|) = O(n (\log n+k))$

<u>Algorithmus für b):</u> für k = 2

- sei $U = [x_1, x_2] \times [y_1, y_2]$ $x_i, y_i \in \Re \cup \{-\infty, \infty\}$
- Suche nach (v, U) liefert alle Punkte in $\text{Reg}(v) \cap U$
- l(v) $\rightarrow$ trennende Gerade
- M(v) $\rightarrow$ Wert der entsprechenden Koordinate
- Suche (v, U):

```
if l(v) vertikal then [l, r] := [x1, x2];
else [l, r] := [y1, y2] ;
if v Blatt then gib zugehörigen Punkt aus, falls ∈ U →
fertig;
else {
if l < M(v) then suche (< -Kind, U);
if l ≤ M(v) ≤ r suche (= -Kind, U);
if r > M(v) then suche (> -Kind, U);
}
```