

NEAREST NEIGHBOR ALGORITHMUS

Gehe mit Sweepline von links nach rechts und speichere in d die kürzeste Distanz zwischen zwei Punkten links von der Sweepline. Initialisiere $d = \infty$

- Verschiebe die Sweepline nach rechts zum nächsten Punkt p .
- Lösche Knoten, die mehr als d Schritte links von p sind.
- Finde Punkt links von p mit kürzester Distanz zu p .
Update d , falls nötig. (Speichere Paar)

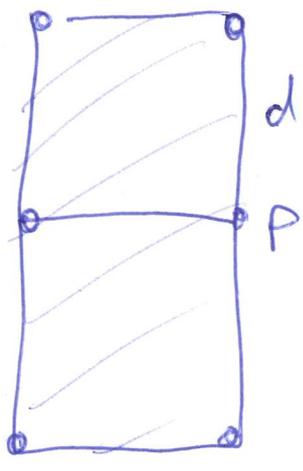
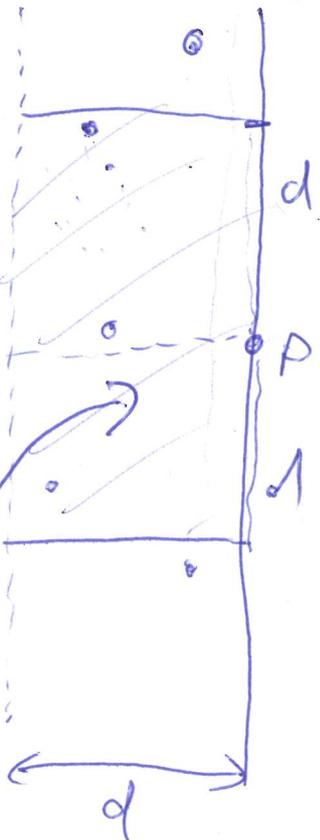
Korrektheit

Jeder Knoten wird mit allen linken Punkten verglichen, die nicht zu weit weg sind. Somit werden alle Paare berücksichtigt.
relevanten.

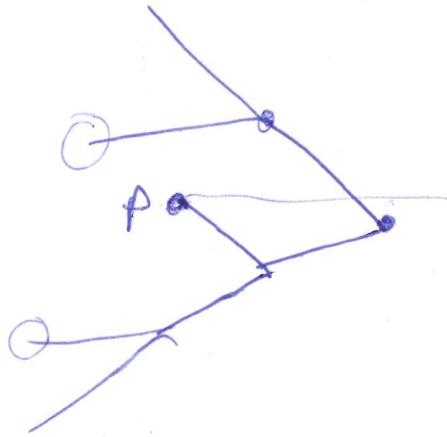
Sweep Line

Worst-Case
6 Punkte in Fläche
d

gelöscht
∞



enthält
maximal
6 Punkte



Laufzeit

Sortiere Knoten von links nach rechts $O(\log n \cdot n)$

Speichere aktive Knoten in balanciertem
Binärbaum

neuen Punkt p einfügen $O(\log n)$

Punkt löschen $O(\log n)$
(maximal n mal)

~~Der~~ Die Punkte im Binärbaum
sind sortiert von oben nach unten.

\Rightarrow Finde obere und untere Nachbarn
von p in $O(\log n)$ (maximal 6 mal)

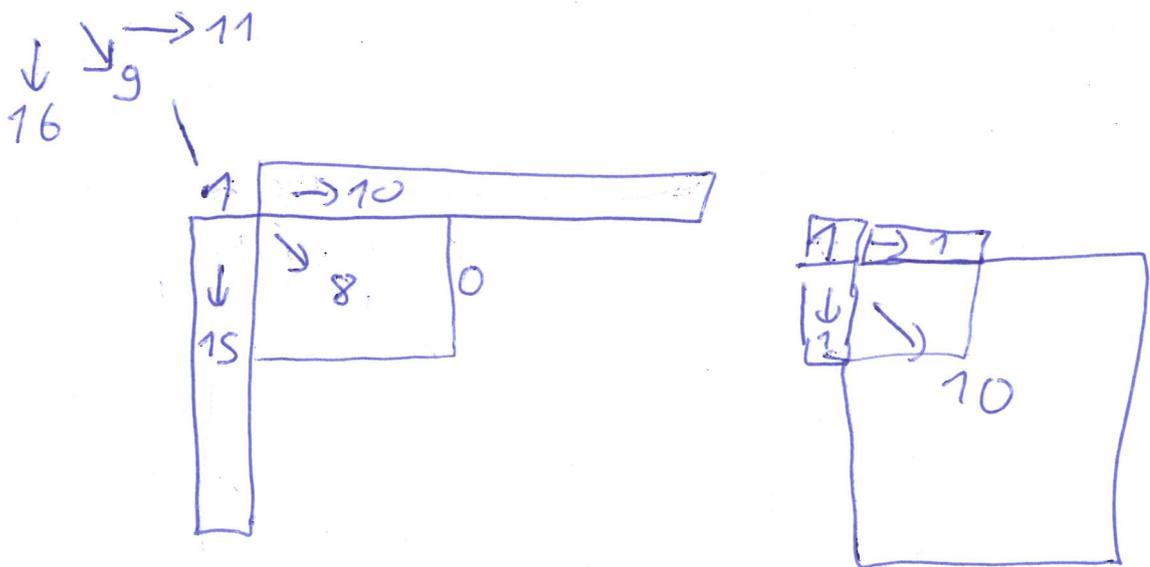
$\Rightarrow O(n \log n)$

Gegeben $n \times n$ 0-1 Matrix

4

$$M = \begin{pmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \\ 0 & 1 & 1 \end{pmatrix}$$

Gesucht: quadratische Submatrix
maximaler Größe mit nur Einsen



$$\downarrow(i,j) = \begin{cases} \text{falls } M(i,j) = 0 \\ 1 + \min(\rightarrow(i,j+1), \downarrow(i+1,j), \downarrow(i+1,j+1)) \\ \text{falls } M(i,j) = 1 \end{cases}$$

$$\rightarrow(i,j) = \begin{cases} 0 & \text{falls } M(i,j) = 0 \\ \rightarrow(i,j+1) + 1 & \text{sonst} \end{cases}$$

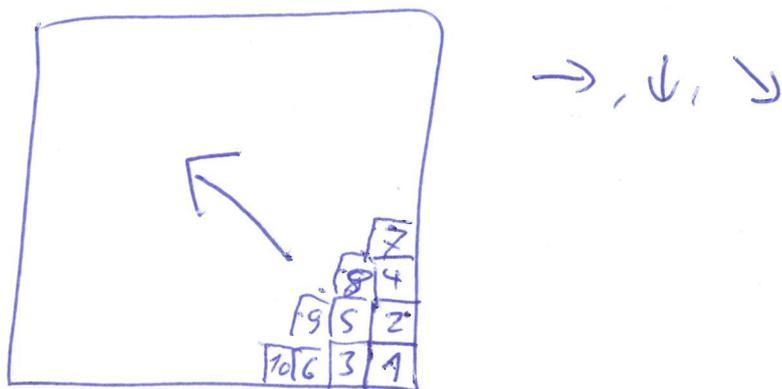
$$\downarrow(i,j) = \begin{cases} 0 & \text{falls } M(i,j) = 0 \\ \downarrow(i+1,j) + 1 & \text{sonst} \end{cases}$$

→ (i, j) : längste Sequenz an Einsen
von (i, j) nach rechts

↓ (i, j) — // — unten

↘ (i, j) Kantenlänge des größten
Quadrats mit nur Einsen und
 (i, j) als oben linker Ecke

Berechnungsreihenfolge von



Antwort: maximaler Eintrag der Matrix ↘

Laufzeit

- Jeder Eintrag von →, ↓, ↘ lässt sich in konstanter Zeit aus den Einträgen unten, rechts und untenrechts berechnen.
 - n^2 Einträge
- ⇒ $O(n^2)$ Laufzeit

nicht bearbeitet:

7

1. Zeichnen Sie einen AVL Baum mit Ebenen und minimal/maximal vielen Knoten.
2. Fügen Sie in einen zu Beginn leeren AVL Baum die Schlüssel 1, 2, 3, 4, 5, 6 in dieser Reihenfolge ein und zeichnen Sie den resultierenden Baum.
3. Beschreiben Sie in eigenen Worten oder Pseudocode, wie das Löschen in einem Splay-Baum funktioniert.
4. Nennen Sie einen Vorteil von Splay-Bäumen gegenüber AVL-Bäumen.
5. Was bedeutet es, dass ein Sortierverfahren stabil ist?
6. Geben Sie ein Sortierverfahren an, das in-place ist und eines das es nicht ist.
7. Zeichnen Sie einen Heap (als Binärbaum), der die Schlüssel 5, 7, 19, 25, 3, 10, 4, 4 enthält.
8. In einem Array, der einen Heap enthält, liegt ein Schlüssel auf Position i . Auf welcher Position liegt dann sein rechtes Kind?
9. Nennen Sie zwei Anwendungen für die Tiefensuche.
10. Welche Vorteile hat die Breitensuche gegenüber der Tiefensuche und umgekehrt?