AlgGeO 1.1

Buch: Algorithmen, Geometrie, TK
2. Auflage
+ einzelne Abschnitte der Vorlesung

Skript: 2.2.3 : Bsp: ① Maximum von n Zahlen

$-3 \quad 2 \quad -1 \quad 5 \mid 11 \quad -20$

$\longrightarrow$ MaxSoFar

② Aktienkurse
Array A
täglicher Kursschwankungen

$-1 \quad 2 \quad -3 \quad \underbrace{5 \quad -2 \quad -1 \quad 8}_{\text{MaxSoFar } 10} \quad -2 \quad 0 \quad 1 \quad -9 \quad \underbrace{3 \quad -1 \quad 3 \quad -4 \quad 8}_{9} \mid ?$

MaxEndingHere

gesucht: maximale Teilsumme konsekutiver Felder

$$\max\left(\max_{1 \le i \le k \le n} \sum_{j=i}^{k} A[j],\ 0\right)$$

einfacher Ansatz

```
MaxSoFar := 0
for i := 1 to n
  for k := i to n
    sum := 0
    for j := i to k
      sum := sum + A[j]
    MaxSoFar := max(MaxSoFar, sum)
```

Laufzeit n^3

AlgGeo 1.2

besserer Ansatz: Summe fortschreiben

```
MaxSoFar := 0
for i := 1 to n
    sum := 0
    for k := i to n
        sum := sum + A[k]
        MaxSoFar := max(MaxSoFar, sum)
```

Laufzeit n^2

Divide & Conquer: Laufzeit $n \log n$

noch besser: Sweep: zur Bestimmung von Eigenschaften von Objekten durchlaufe Objekte von links nach rechts ~~sch~~ unterhalte die Eigenschaft der bisher besuchten Objekte stelle fest, welche Ereignisse die Eigenschaft verändern und was zu tun ist, um sie wiederherzustellen

Kochrezept

☹ Müssen nicht nur MaxSoFar, sondern auch MaxEndingHere unterhalten

☺ Wie kann sich MaxEndingHere verändern?

$$\underbrace{[\underbrace{3 \quad -1 \quad 3 \quad -4 \quad 8}_{9} \mid -7}_{2}$$

↑ bleibt

$$\underbrace{[\underbrace{3 \quad -1 \quad 3 \quad -4 \quad 8}_{9} \mid 2]}_{11}$$

AlgGeo 1.3 $[\underbrace{3 \quad -1 \quad 3 \quad -4 \quad 8}_{9} \mid -10$ $\quad 3 \quad -1 \quad 3 \quad -4 \quad 8 \quad -10 \; \lfloor$

MaxEndingHere := 0

<u>optimal</u>

```
MaxSoFar := 0
MaxEndingHere := 0
for i := 1 to n
    MaxEndingHere := max(MaxEndingHere + A[i], 0)
    MaxSoFar      := max(MaxSoFar, MaxEndingHere)
```

$O(n)$

John Bentley: Programming Pearls

<u>Sheep in 2D</u> Closest Pair Problem

Gegeben: n Punkte in der Ebene

Gesucht: kleinster Abstand zwischen zwei Punkte

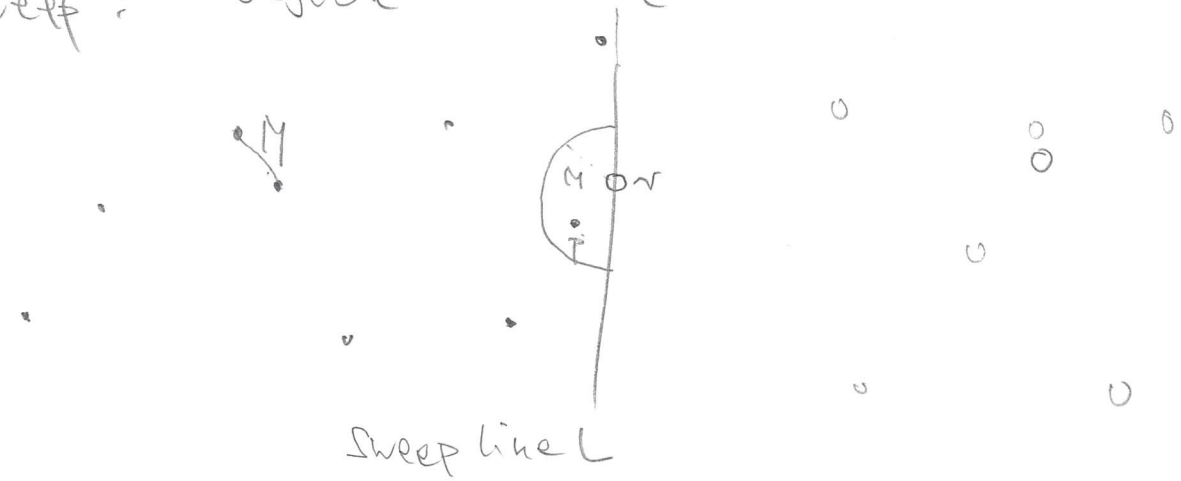
Algorithmus: kennt Koordinaten $(x_p, y_p) = p$

kann euklidische Abstände ausrechnen

$$|p-q| = \sqrt{(x_p - x_q)^2 + (y_p - y_q)^2}$$

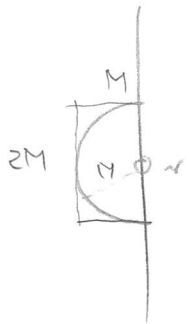
naiv: alle $\binom{n}{2}$ Abstände berechnen, Minimum bestimmen | $\binom{n}{2} \approx n^2$

Sweep? Objekte: Punkte

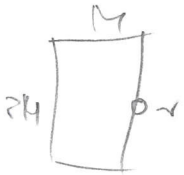


Beobachtung: Wenn unser Punkt r von Sweepline L getroffen wird (Event!), kann er nur mit Partnerpunkten p im linken Halbkreis um r mit Radius M das alte Minimum M unterbieten

☹ Kreisanfragen sind schwierig:



Idee: Vergröberung:
statt Halbkreis: Box



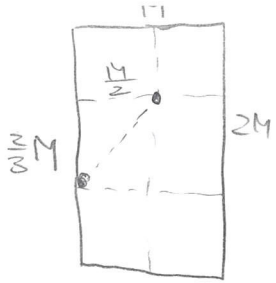
teste alle Punkte p in Box, ob sie zu r Abstand < M haben

2 Probleme: (i) wieviele Punkte können in der Box liegen (Vermutung: 6)
(ii) wie findet man sie?

AlgGeo 1.5

zu (i):

Lemma: In einer Box



können maximal 6 Punkte mit paarweisem Abstand $\geq M$ liegen.

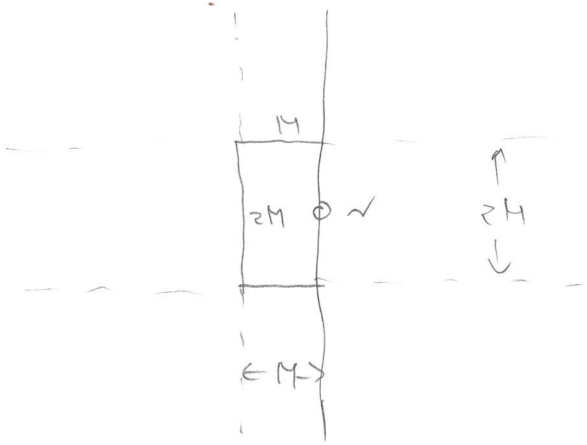
Bew: In jeder Minibox kann höchstens 1 Punkt liegen.

Denn: 2 Punkte in einer Minibox haben höchstens den Abstand

$$\sqrt{\left(\frac{2}{3}M\right)^2 + \left(\frac{M}{2}\right)^2} = \underbrace{\sqrt{\frac{16+9}{36}}}_{\frac{5}{6}} M < M.$$

$\boxed{\text{Lemma}}$

zu (ii): Wie berichtet man die Punkte in der Box?



Idee: Unterhalte alle Punkte im Streifen der Breite M links von Sweepline L, nach y sortiert.

Wenn L auf neuen Punkt $r = (x_r, y_r)$ stößt:

berichte alle Punkte p im Streifen mit y-Koordinate $y_r - M \leq y_p \leq y_r + M$.

AlgGeo 1.6

Benötigen Datenstruktur ~~mit~~ y-Werte für folgende Operationen:

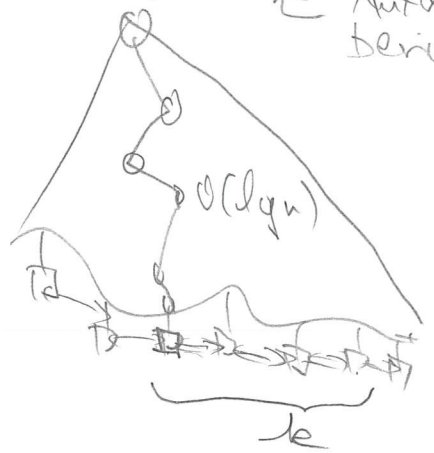
Einfügen, Entfernen $O(\log n)$

Berichten aller Werte im Intervall $[y_1, y_2]$

Black Box:

AVL-Bäume

$O(\log n + k)$ ← Anzahl der berichteten Punkte



Algorithmus für Closest Pair Problem im $\mathbb{R}^2$

Sortiere Punkte nach x-Koord. $O(n \log n)$

bewege Sweepline L von links nach rechts

wenn neuer Punkt r getroffen:

entferne alle Punkte q mit $x_q < y_r - M$ aus Streifen

$\boxed{\text{insgesamt } O(n \log n)}$

Sweepline L hält n mal an

finde alle Punkte p im Streifen mit

$y_r - M \leq y_p \leq y_r + M$ $O(\log n + \underbrace{k}_{\leq 6})$

teste, ob für ein solches p gilt: $O(1)$

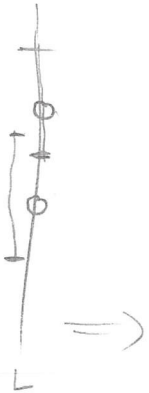
$|p - r| < M$

falls ja: $M := |p - r|$

füge r in Streifen ein: $O(\log n)$

Theorem 2.4 Der dichteste Abstand von n Punkten im $\mathbb{R}^2$ läßt sich in Zeit $O(n \log n)$ bestimmen.

Frage: Was passiert hier:



1. Möglichkeit
   Algorithmus anpassen

2. Möglichkeit
   "Allgemeine Lage" verlangen