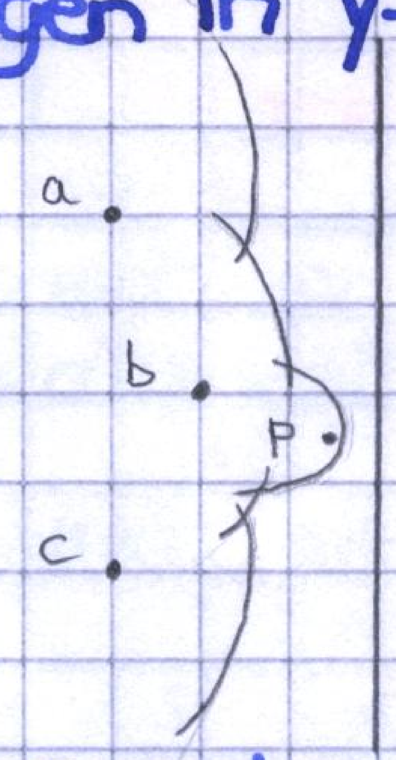
3.3.3 Details der Implementierung

- **Site-Event:**
  - Finde $P(a,b,c)$ (in der Y-Struktur) der von horizontaler Geraden durch p geschnitten wird
  - Lösche $P(a,b,c)$ aus Y-Struktur
  - Lösche Circle-Event von $P(a,b,c)$ aus X-Struktur
  - Füge 3 neue Bögen in Y-Struktur ein:
    - $P(a,b,p)$
    - $P(b,p,b)$
    - $P(p,b,c)$



  - Berechne Circle-Events der neuen Parabelstücke und füge sie in die X-Struktur ein.

- **Circle-Event:**
  - Gib Mittelpkt des Kreises durch a, b, c als Voronoi-Knoten aus und Orte a, b, c auf dem Kreis gegen den UZS.
  - Seien d und e die Orte deren Parabelbögen mit denen von a und c benachbart sind.
    - $P(d,a,b)$ wird zu $P(d,a,c)$
    - $P(b,c,e)$ wird zu $P(a,c,e)$

    } Änderung in Y-Struktur
  - Circle-Events von $P(d,a,b)$ und $P(b,c,e)$ löschen und Circle-Event von $P(d,a,c)$ und $P(a,c,e)$ einfügen.

- **Satz:**
  Das VD von n pktförmigen Orten kann durch Plane Sweep in Zeit $O(n \log n)$ berechnet werden.

  Beweis:
  Alle Operationen auf X- bzw Y-Struktur können in Zeit $O(\log n)$ ausgeführt werden wenn sie durch binäre Suchbäume realisiert werden. Außerdem hat das VD lineare Größe in der Zahl der Orte.

- **Bemerkung:**
  Algorithmus kann für andere Formen von Orten verallgemeinert werden. Dazu muss man kompliziertere "Bisektoren" betrachten.
  - 2 Pkte → Bisektor ist Mittelsenkrechte
  - 1 Pkt / 1 Gerade → —— " —— Parabelbogen
  - 2 Geraden → —— " —— Winkelhalbierende

3.3.4 Point Location

- **Aufgabe:**
  Finde für bel Pkt $p \in \mathbb{R}^2$ die VR die p enthält.
  Dann ist der Ort dieser Region der der von allen anderen p am nächsten liegt.

$M := \#$ Abfragen
- **Idee der Vorverarbeitung:** gilt nur für Streifenmethode
  Bezahle $O(n \log n)$ für Konstruktion von VD
  ⇒ Laufzeit $O(\log n)$ pro Anfrage (sehr effizient bei vielen Anfragen)
  Sonst Minimumsuche über Distanzen ⇒ kostet $\underbrace{O(n)}_{\text{schlechter als } O(\log n)\,!!}$!!

Falls $M < n \Rightarrow O(M \cdot n) \cap O(n \log n) \Rightarrow O(M \cdot n) = O(n)$ besser
Falls $M > n \Rightarrow O(n^2) \cap O(M \log n) \Rightarrow O(M \log n)$ besser