

Algorithmische Geometrie

Übung 3



Blatt 2 – Allgemeines

Blatt 2 – Allgemeines

- gut: oft erst Idee erklärt

Blatt 2 – Allgemeines

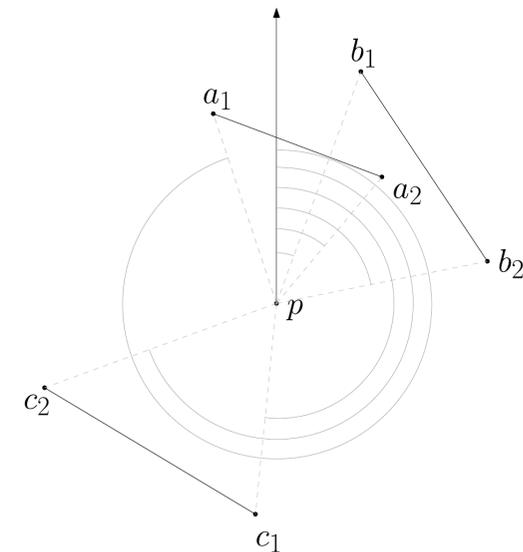
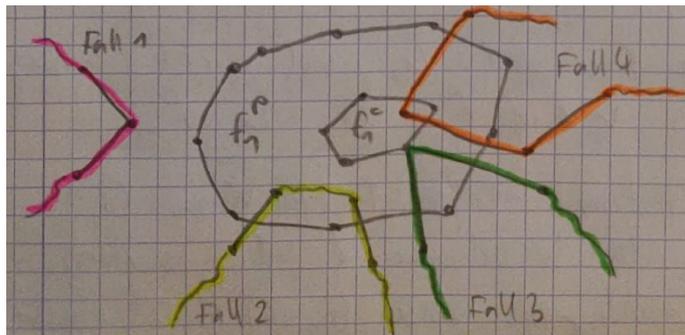
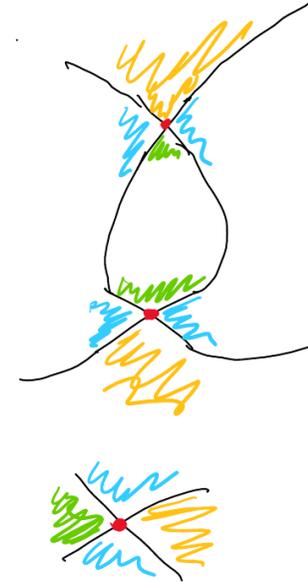
- gut: oft erst Idee erklärt
- spezifiziert Algorithmen vollständig

Blatt 2 – Allgemeines

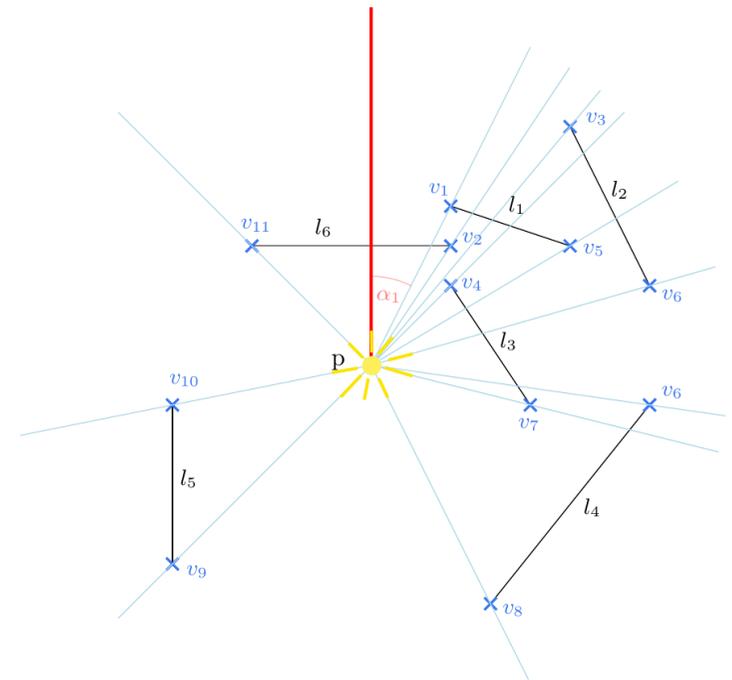
- gut: oft erst Idee erklärt
- spezifiziert Algorithmen vollständig
- z.B. Sweepline
 - Initialisierung
 - Events (Sortierung?)
 - Status (Sortierung?)
 - Events verarbeiten (unterschiedliche Typen)

Blatt 2 – Allgemeines

- gut: oft erst Idee erklärt
- spezifiziert Algorithmen vollständig
- z.B. Sweepline
 - Initialisierung
 - Events (Sortierung?)
 - Status (Sortierung?)
 - Events verarbeiten (unterschiedliche Typen)



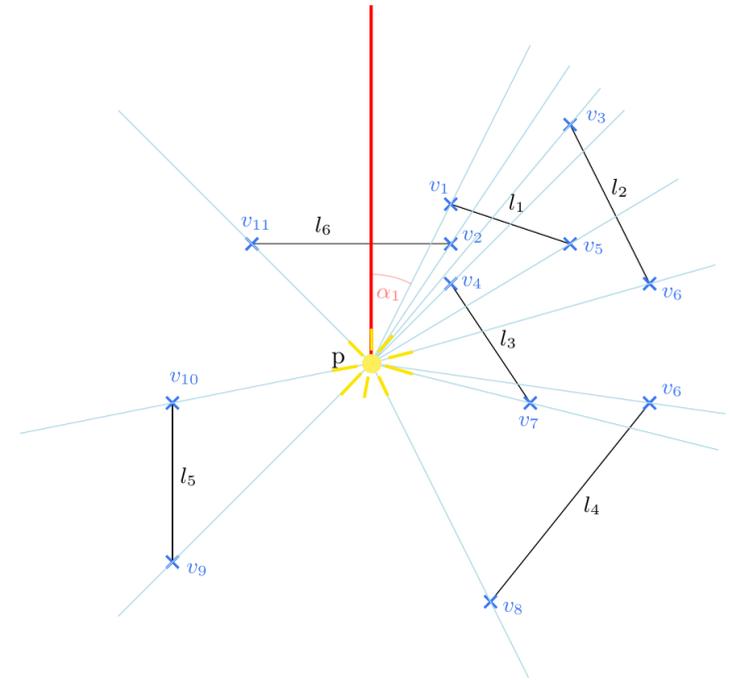
Blatt 2 – Aufgabe 1



Blatt 2 – Aufgabe 1

Sweepeline

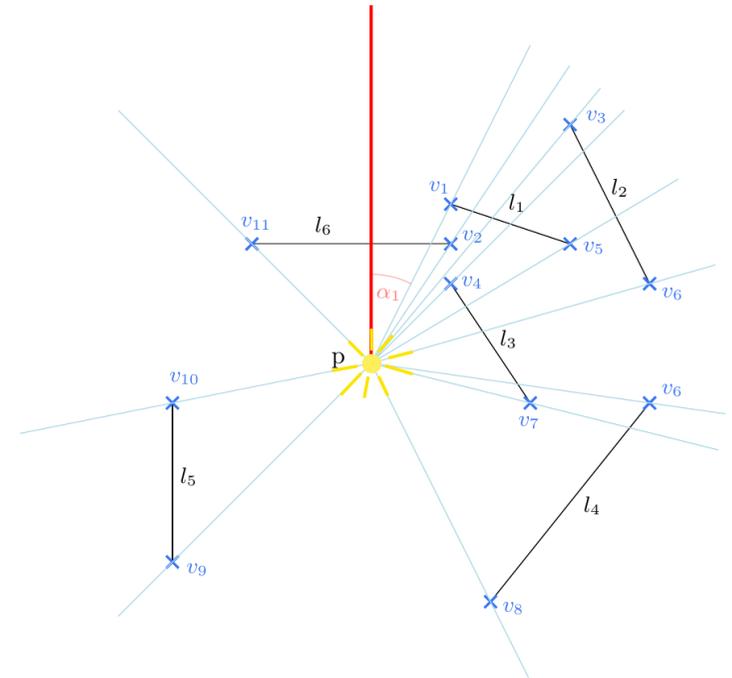
- Events: Start- und Endpunkte



Blatt 2 – Aufgabe 1

Sweepeline

- Events: Start- und Endpunkte
- Status: geschnittene Wände

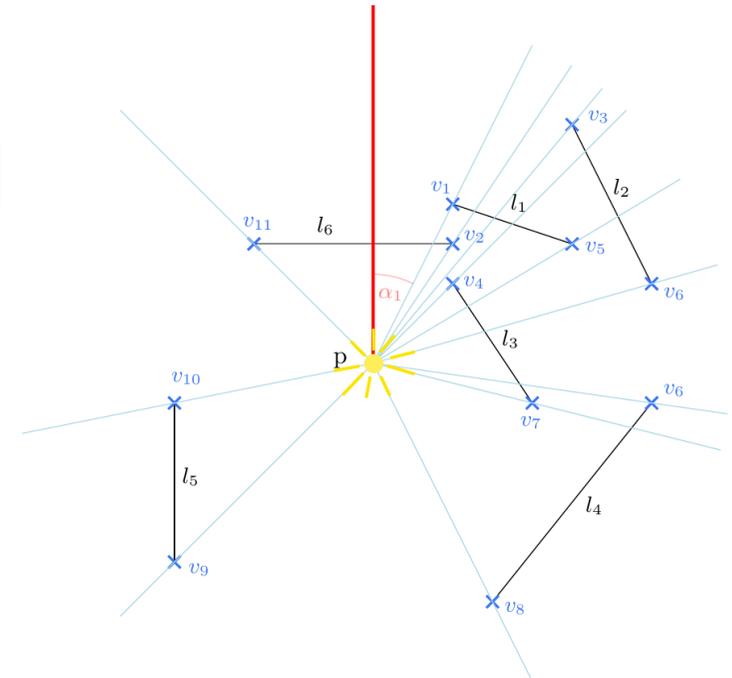


Blatt 2 – Aufgabe 1

Sweepeline

- Events: Start- und Endpunkte
- Status: geschnittene Wände

Sortierung?



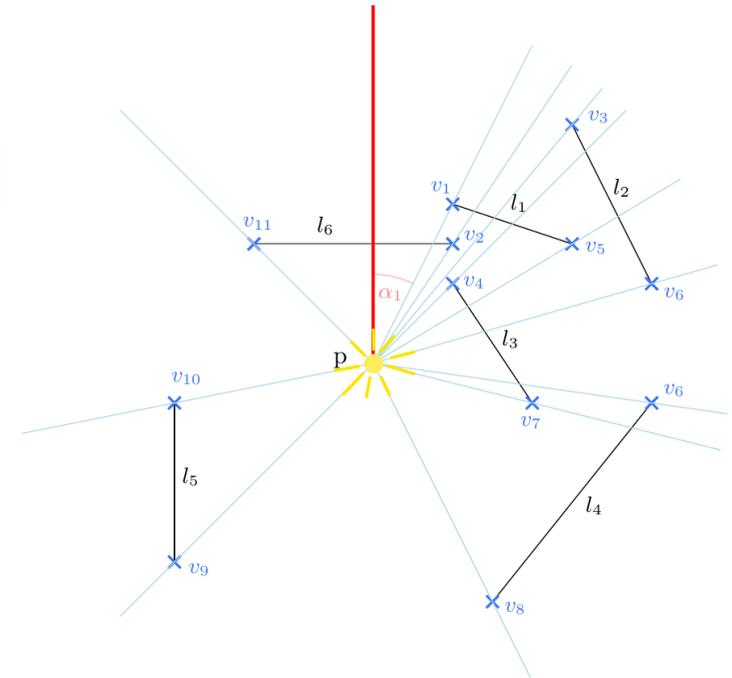
Blatt 2 – Aufgabe 1

Sweepeline

- Events: Start- und Endpunkte
- Status: geschnittene Wände

- Events verarbeiten

Sortierung?



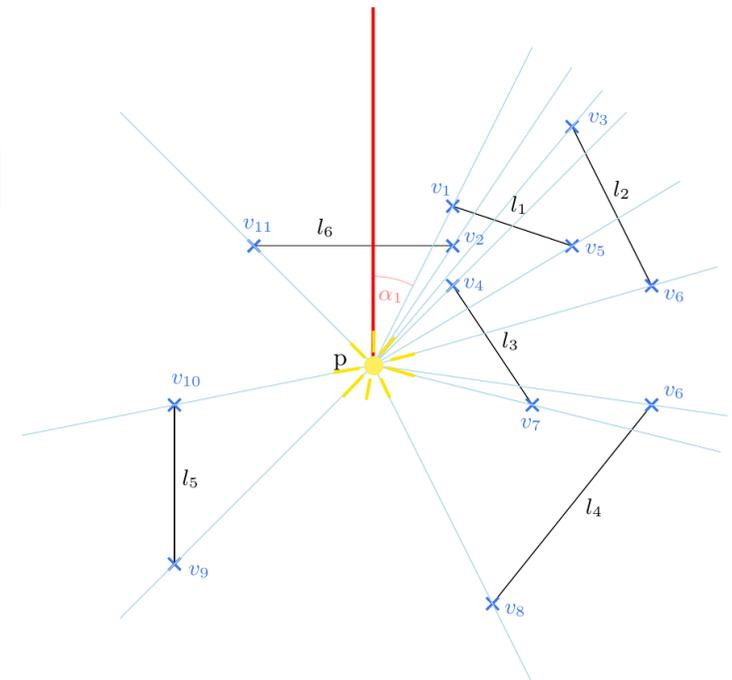
Blatt 2 – Aufgabe 1

Sweepeline

- Events: Start- und Endpunkte
- Status: geschnittene Wände

- Events verarbeiten
 - Startpunkt: Wand in Status einfügen

Sortierung?



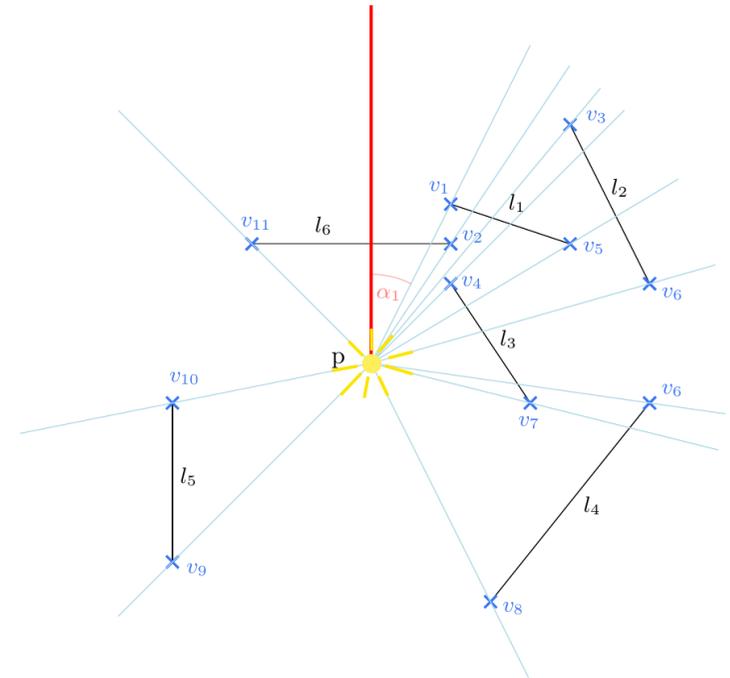
Blatt 2 – Aufgabe 1

Sweepeline

- Events: Start- und Endpunkte
- Status: geschnittene Wände

- Events verarbeiten
 - Startpunkt: Wand in Status einfügen
 - Endpunkt: Wand aus Status löschen

Sortierung?



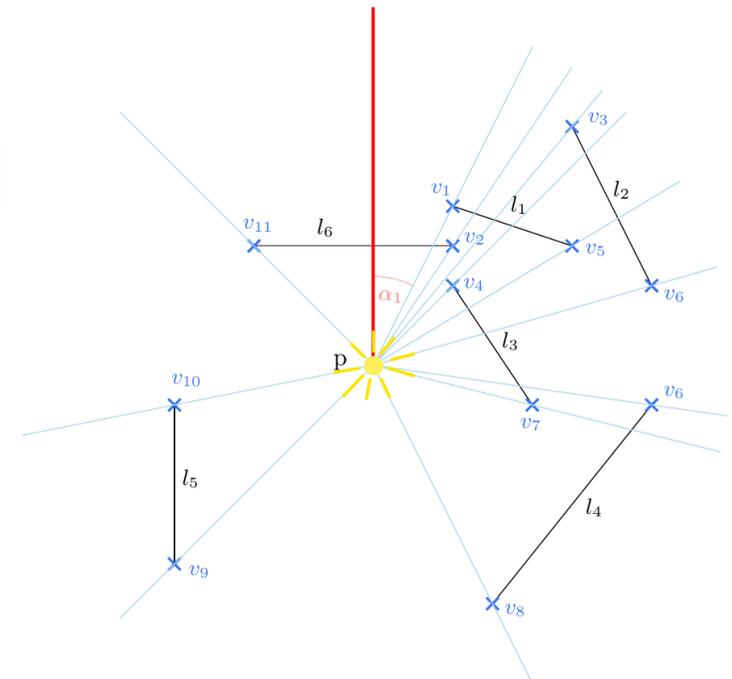
Blatt 2 – Aufgabe 1

Sweepeline

- Events: Start- und Endpunkte
- Status: geschnittene Wände

- Events verarbeiten
 - Startpunkt: Wand in Status einfügen
 - Endpunkt: Wand aus Status löschen
 - erste Wand markieren

Sortierung?



Blatt 2 – Aufgabe 1

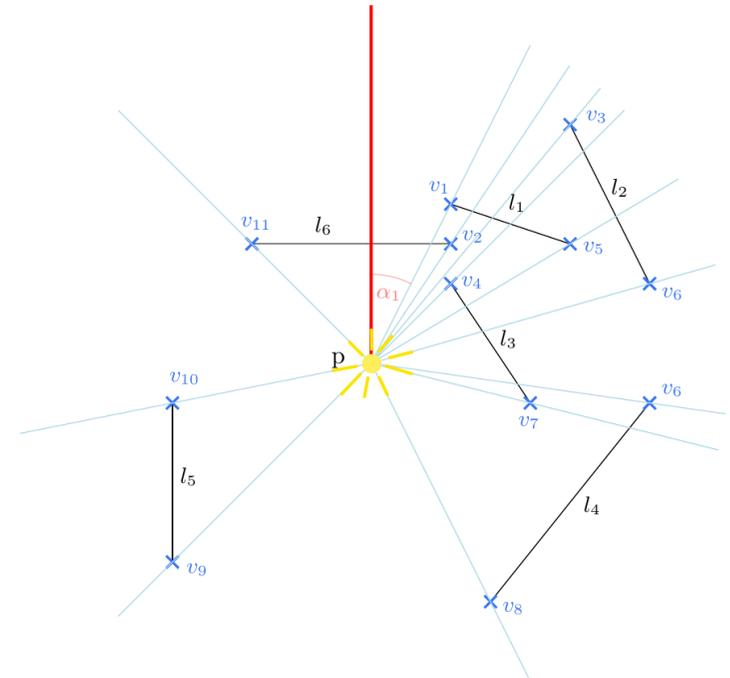
Sweepeline

- Events: Start- und Endpunkte
- Status: geschnittene Wände

- Events verarbeiten
 - Startpunkt: Wand in Status einfügen
 - Endpunkt: Wand aus Status löschen
 - erste Wand markieren

Sortierung?

Sonderfälle?



Blatt 2 – Aufgabe 1

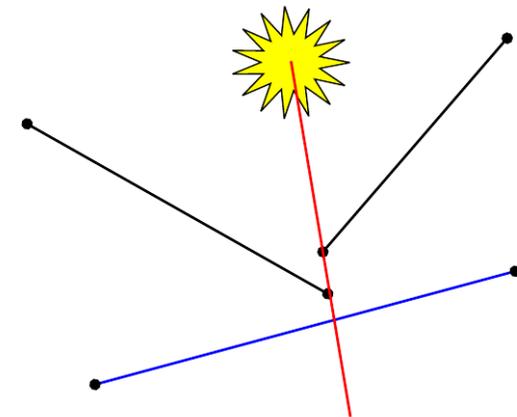
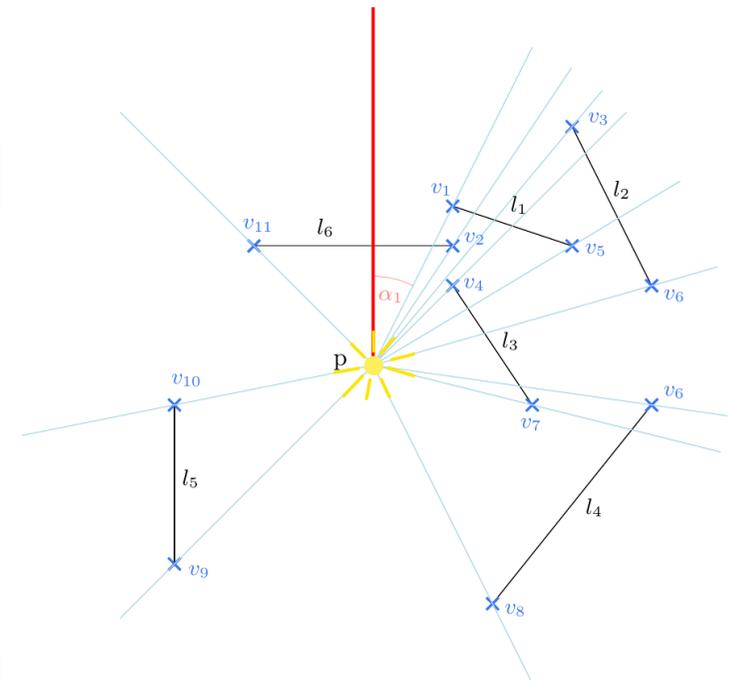
Sweepeline

- Events: Start- und Endpunkte
- Status: geschnittene Wände

- Events verarbeiten
 - Startpunkt: Wand in Status einfügen
 - Endpunkt: Wand aus Status löschen
 - erste Wand markieren

Sortierung?

Sonderfälle?



Blatt 2 – Aufgabe 1

Sweepeline

- Events: Start- und Endpunkte
- Status: geschnittene Wände

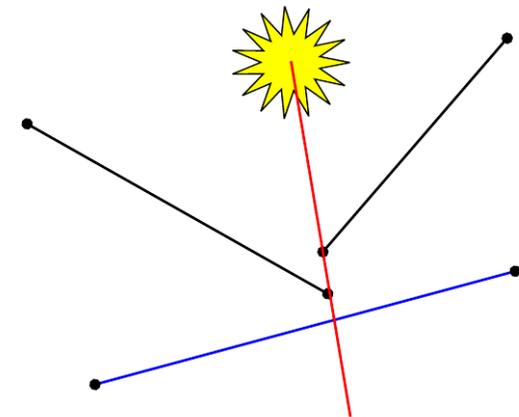
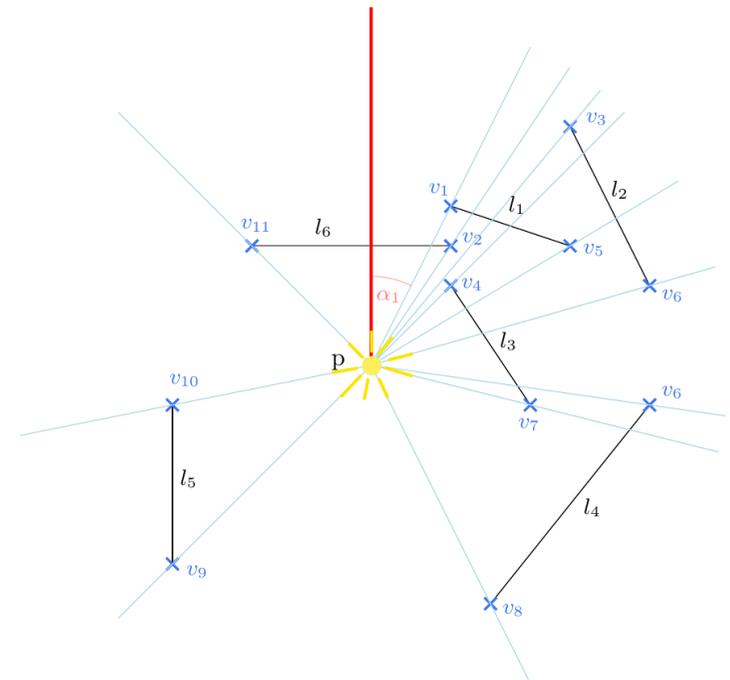
- Events verarbeiten
 - Startpunkt: Wand in Status einfügen
 - Endpunkt: Wand aus Status löschen
 - erste Wand markieren

- Initialisierung

Sortierung?

Sonderfälle?

Wie?



Blatt 2 – Aufgabe 1

Sweepeline

- Events: Start- und Endpunkte
- Status: geschnittene Wände

- Events verarbeiten
 - Startpunkt: Wand in Status einfügen
 - Endpunkt: Wand aus Status löschen
 - erste Wand markieren

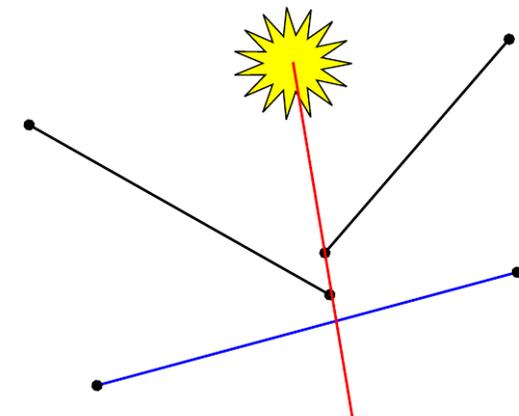
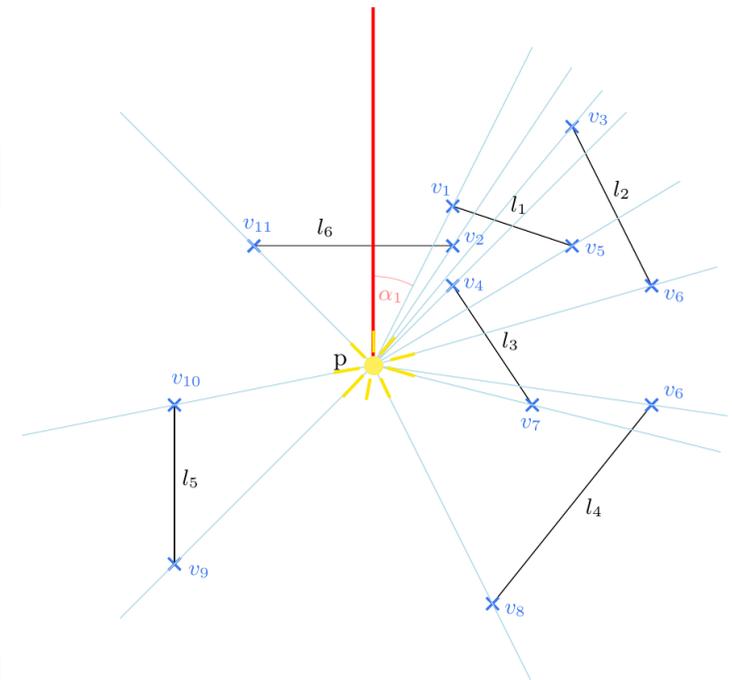
Sortierung?

Sonderfälle?

Wie?

- Initialisierung

- Laufzeit?



Blatt 2 – Aufgabe 2

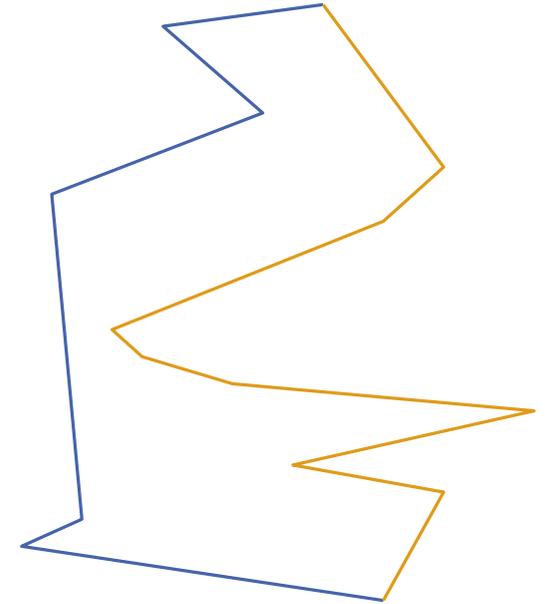
Trianguliere y -monotones Polygon

- sortiere Punkte links und rechts nach y -Koordinate

Blatt 2 – Aufgabe 2

Trianguliere y -monotones Polygon

- sortiere Punkte links und rechts nach y -Koordinate



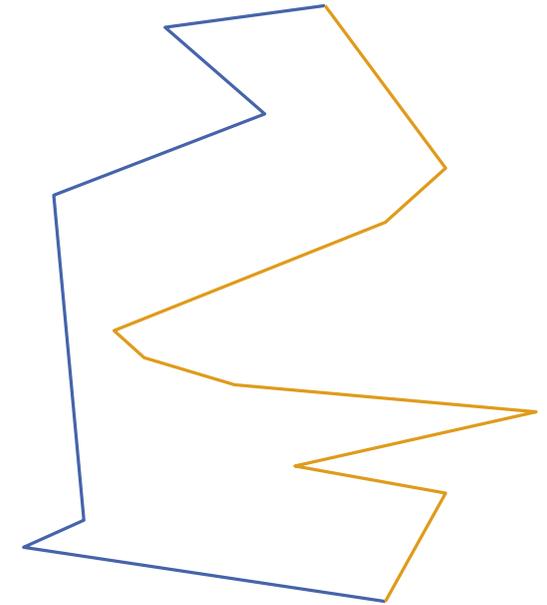
Blatt 2 – Aufgabe 2

Trianguliere y -monotones Polygon

- sortiere Punkte links und rechts nach y -Koordinate

Was funktioniert?

- verbinde im Zickzack alle Ecken von oben nach unten abwechselnd miteinander



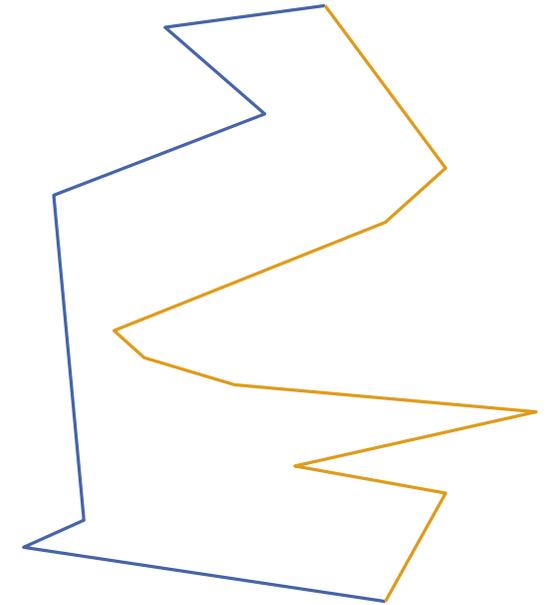
Blatt 2 – Aufgabe 2

Trianguliere y-monotones Polygon

- sortiere Punkte links und rechts nach y-Koordinate

Was funktioniert?

- verbinde im Zickzack alle Ecken von oben nach unten abwechselnd miteinander
- verbinde immer die zwei benachbarten Punkte vom höchsten Punkt



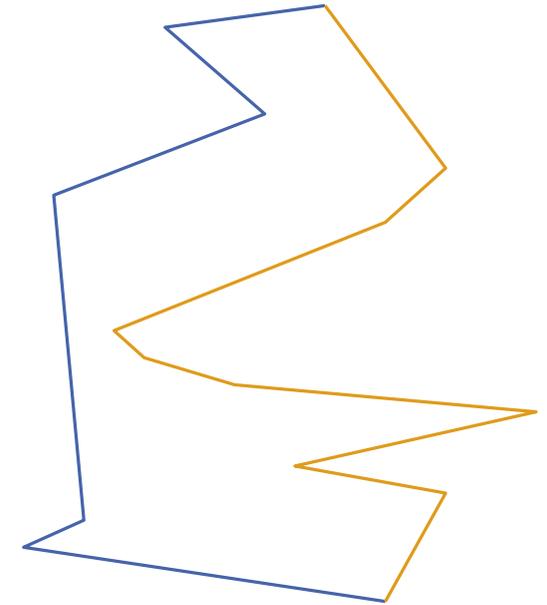
Blatt 2 – Aufgabe 2

Trianguliere y -monotones Polygon

- sortiere Punkte links und rechts nach y -Koordinate

Was funktioniert?

- verbinde im Zickzack alle Ecken von oben nach unten abwechselnd miteinander
- verbinde immer die zwei benachbarten Punkte vom höchsten Punkt
- immer Ohr um aktuellen Punkt abschneiden



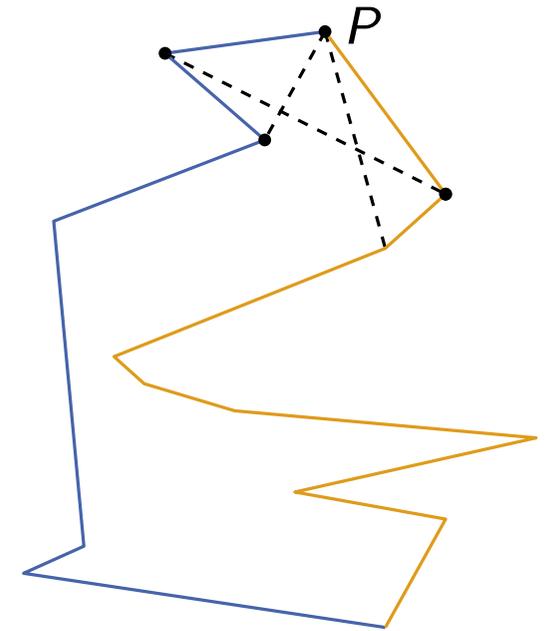
Blatt 2 – Aufgabe 2

Trianguliere y -monotones Polygon

- sortiere Punkte links und rechts nach y -Koordinate

Was funktioniert?

- verbinde im Zickzack alle Ecken von oben nach unten abwechselnd miteinander
- verbinde immer die zwei benachbarten Punkte vom höchsten Punkt
- immer Ohr um aktuellen Punkt abschneiden



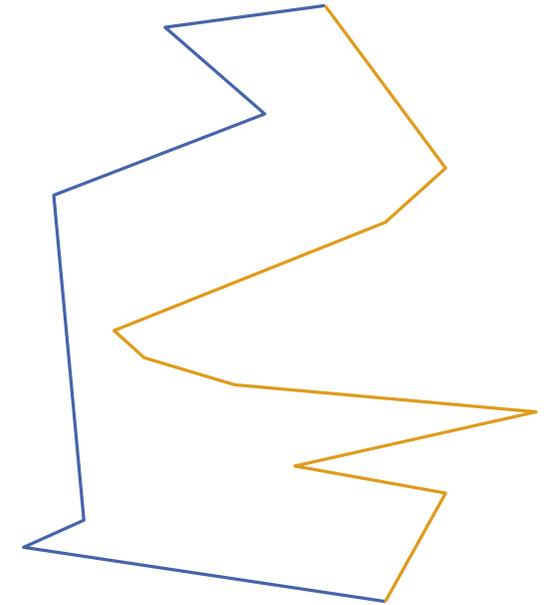
Blatt 2 – Aufgabe 2

Trianguliere y -monotones Polygon

- sortiere Punkte links und rechts nach y -Koordinate

Was funktioniert?

- verbinde im Zickzack alle Ecken von oben nach unten abwechselnd miteinander
- verbinde immer die zwei benachbarten Punkte vom höchsten Punkt
- immer Ohr um aktuellen Punkt abschneiden
- in Komponenten aufteilen und Komponenten triangulieren



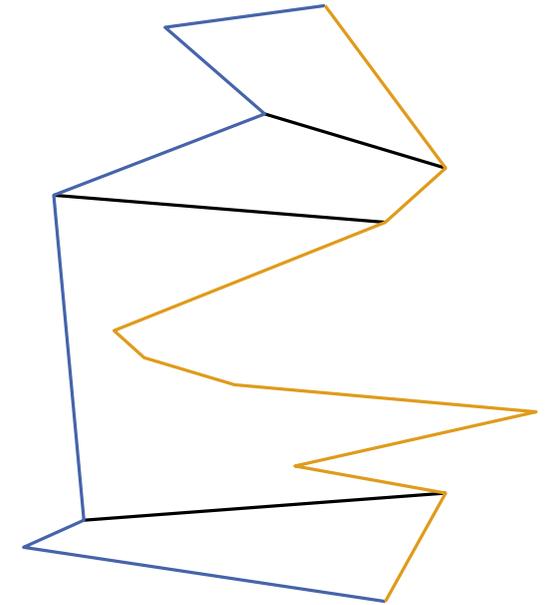
Blatt 2 – Aufgabe 2

Trianguliere y -monotones Polygon

- sortiere Punkte links und rechts nach y -Koordinate

Was funktioniert?

- verbinde im Zickzack alle Ecken von oben nach unten abwechselnd miteinander
- verbinde immer die zwei benachbarten Punkte vom höchsten Punkt
- immer Ohr um aktuellen Punkt abschneiden
- in Komponenten aufteilen und Komponenten triangulieren
 - auf einer Seite genau zwei Punkte



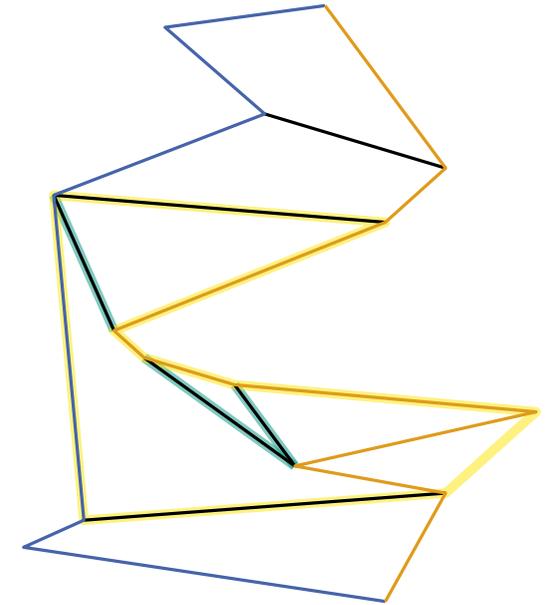
Blatt 2 – Aufgabe 2

Trianguliere y-monotones Polygon

- sortiere Punkte links und rechts nach y-Koordinate

Was funktioniert?

- verbinde im Zickzack alle Ecken von oben nach unten abwechselnd miteinander
- verbinde immer die zwei benachbarten Punkte vom höchsten Punkt
- immer Ohr um aktuellen Punkt abschneiden
- in Komponenten aufteilen und Komponenten triangulieren
 - auf einer Seite genau zwei Punkte
 - konvexe Segmente triangulieren



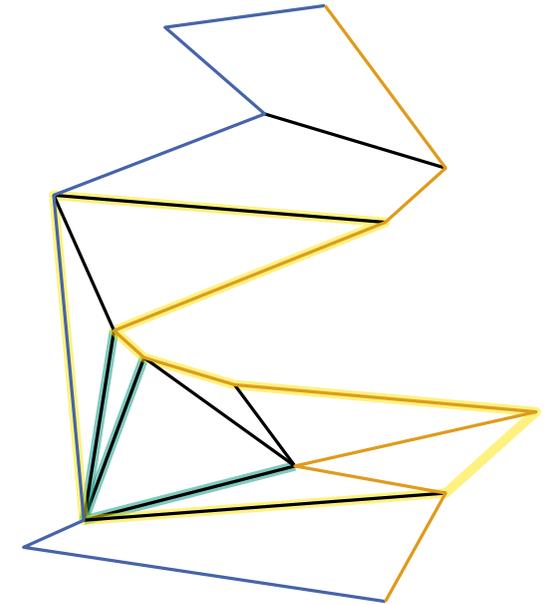
Blatt 2 – Aufgabe 2

Trianguliere y-monotones Polygon

- sortiere Punkte links und rechts nach y-Koordinate

Was funktioniert?

- verbinde im Zickzack alle Ecken von oben nach unten abwechselnd miteinander
- verbinde immer die zwei benachbarten Punkte vom höchsten Punkt
- immer Ohr um aktuellen Punkt abschneiden
- in Komponenten aufteilen und Komponenten triangulieren
 - auf einer Seite genau zwei Punkte
 - konvexe Segmente triangulieren
 - Rest mit einem Punkt auf der anderen Seite verbinden



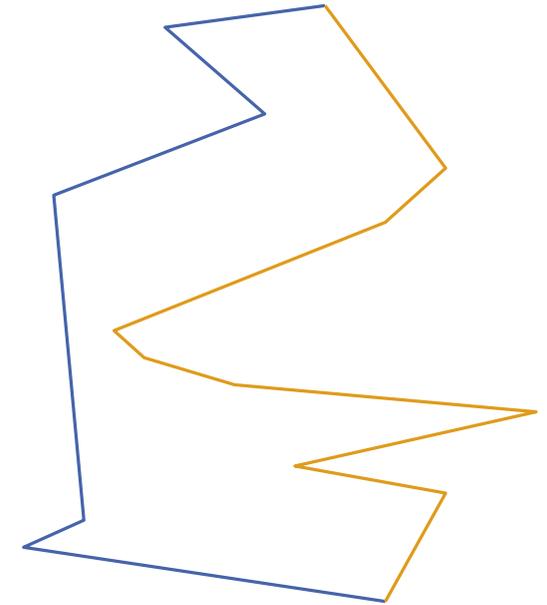
Blatt 2 – Aufgabe 2

Trianguliere y -monotones Polygon

- sortiere Punkte links und rechts nach y -Koordinate

Was funktioniert?

- verbinde im Zickzack alle Ecken von oben nach unten abwechselnd miteinander
- verbinde immer die zwei benachbarten Punkte vom höchsten Punkt
- immer Ohr um aktuellen Punkt abschneiden
- in Komponenten aufteilen und Komponenten triangulieren
- Punkte von oben nach unten abarbeiten, greedy Diagonale



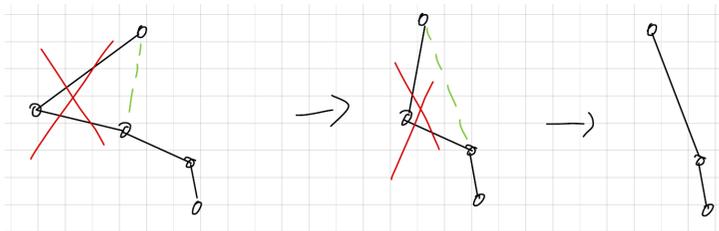
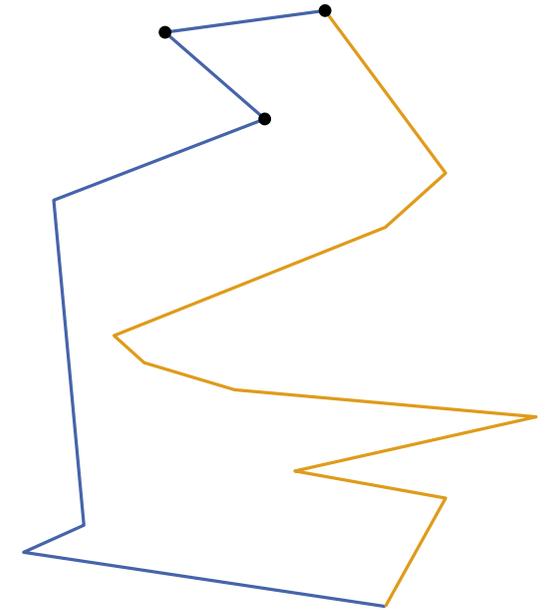
Blatt 2 – Aufgabe 2

Trianguliere y-monotones Polygon

- sortiere Punkte links und rechts nach y-Koordinate

Was funktioniert?

- verbinde im Zickzack alle Ecken von oben nach unten abwechselnd miteinander
- verbinde immer die zwei benachbarten Punkte vom höchsten Punkt
- immer Ohr um aktuellen Punkt abschneiden
- in Komponenten aufteilen und Komponenten triangulieren
- Punkte von oben nach unten abarbeiten, greedy Diagonale
 - auf gleicher Seite: Diagonale falls passende Kurve



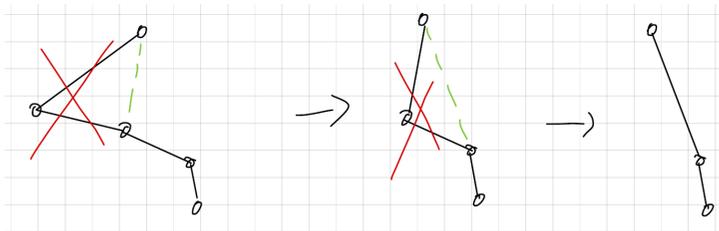
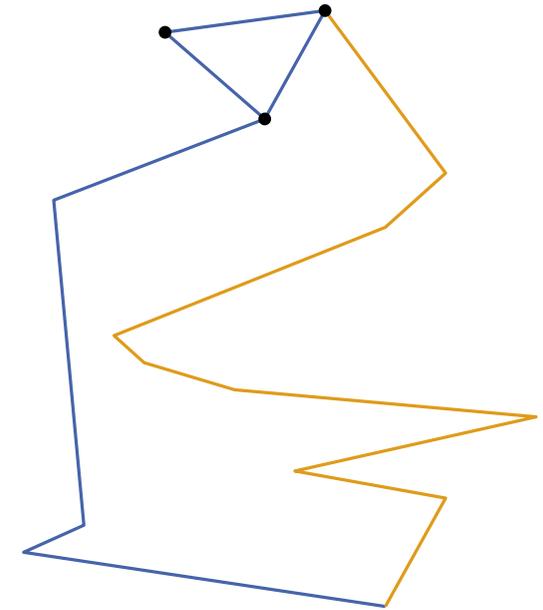
Blatt 2 – Aufgabe 2

Trianguliere y-monotones Polygon

- sortiere Punkte links und rechts nach y-Koordinate

Was funktioniert?

- verbinde im Zickzack alle Ecken von oben nach unten abwechselnd miteinander
- verbinde immer die zwei benachbarten Punkte vom höchsten Punkt
- immer Ohr um aktuellen Punkt abschneiden
- in Komponenten aufteilen und Komponenten triangulieren
- Punkte von oben nach unten abarbeiten, greedy Diagonale
 - auf gleicher Seite: Diagonale falls passende Kurve



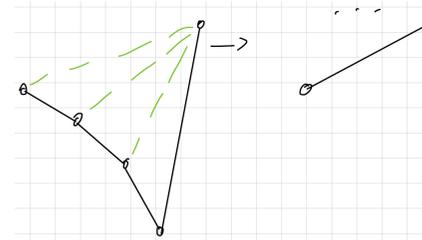
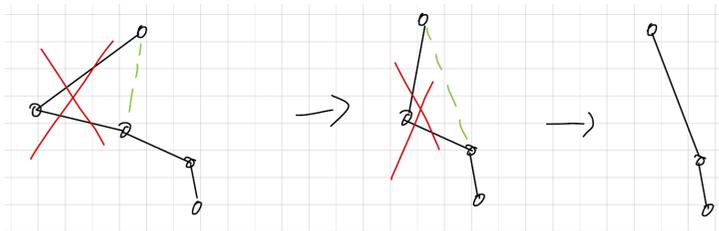
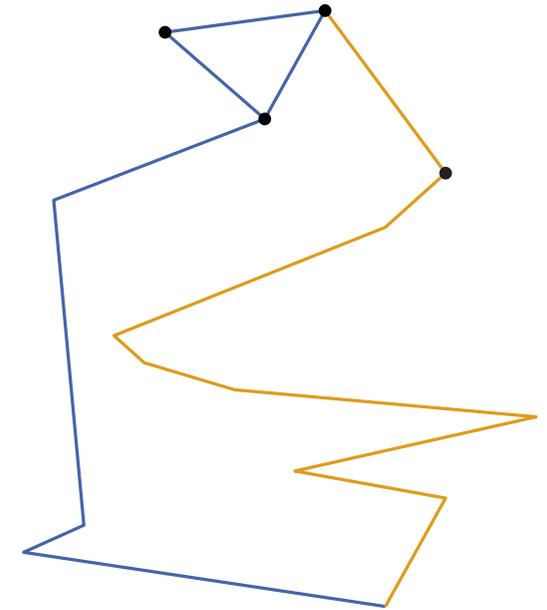
Blatt 2 – Aufgabe 2

Trianguliere y-monotones Polygon

- sortiere Punkte links und rechts nach y-Koordinate

Was funktioniert?

- verbinde im Zickzack alle Ecken von oben nach unten abwechselnd miteinander
- verbinde immer die zwei benachbarten Punkte vom höchsten Punkt
- immer Ohr um aktuellen Punkt abschneiden
- in Komponenten aufteilen und Komponenten triangulieren
- Punkte von oben nach unten abarbeiten, greedy Diagonale
 - auf gleicher Seite: Diagonale falls passende Kurve
 - unterschiedliche Seiten: alle Diagonalen einfügen



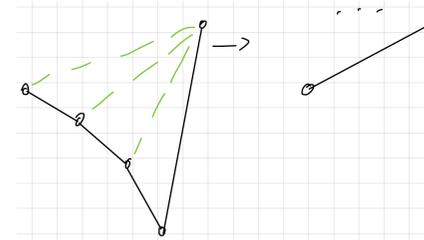
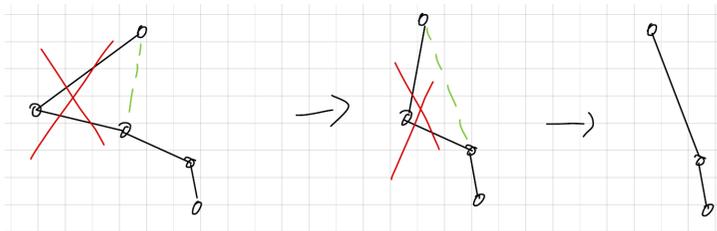
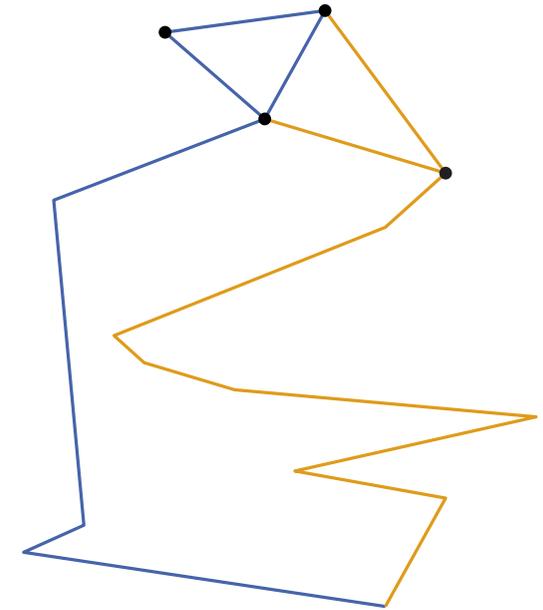
Blatt 2 – Aufgabe 2

Trianguliere y-monotones Polygon

- sortiere Punkte links und rechts nach y-Koordinate

Was funktioniert?

- verbinde im Zickzack alle Ecken von oben nach unten abwechselnd miteinander
- verbinde immer die zwei benachbarten Punkte vom höchsten Punkt
- immer Ohr um aktuellen Punkt abschneiden
- in Komponenten aufteilen und Komponenten triangulieren
- Punkte von oben nach unten abarbeiten, greedy Diagonale
 - auf gleicher Seite: Diagonale falls passende Kurve
 - unterschiedliche Seiten: alle Diagonalen einfügen



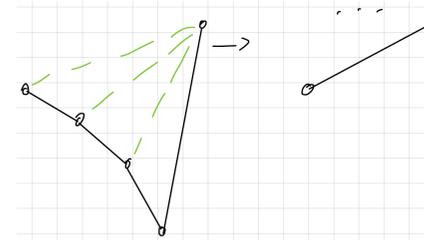
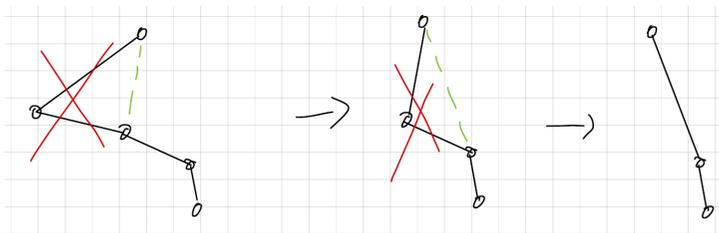
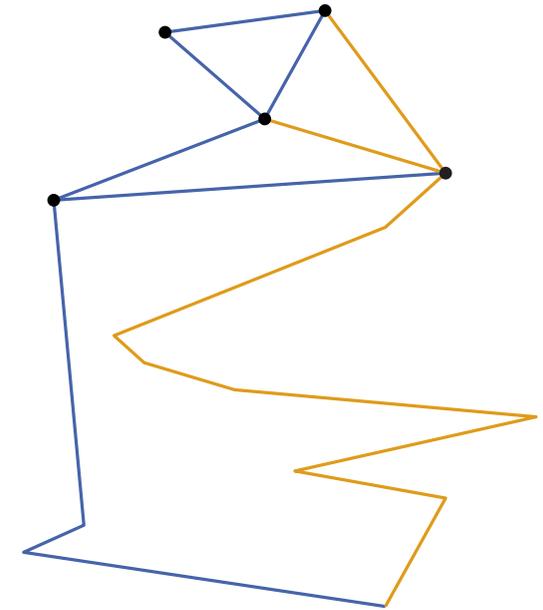
Blatt 2 – Aufgabe 2

Trianguliere y-monotones Polygon

- sortiere Punkte links und rechts nach y-Koordinate

Was funktioniert?

- verbinde im Zickzack alle Ecken von oben nach unten abwechselnd miteinander
- verbinde immer die zwei benachbarten Punkte vom höchsten Punkt
- immer Ohr um aktuellen Punkt abschneiden
- in Komponenten aufteilen und Komponenten triangulieren
- Punkte von oben nach unten abarbeiten, greedy Diagonale
 - auf gleicher Seite: Diagonale falls passende Kurve
 - unterschiedliche Seiten: alle Diagonalen einfügen



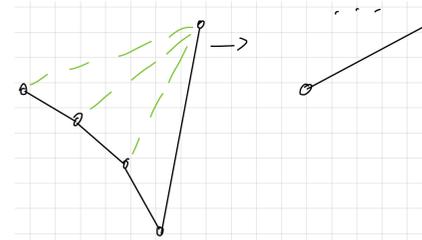
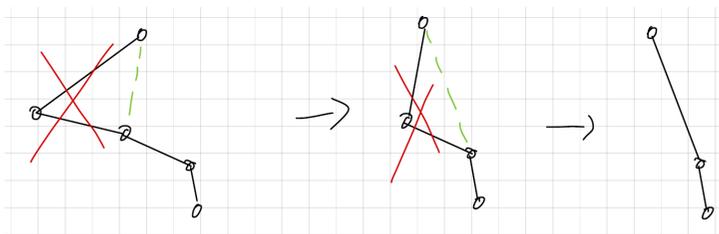
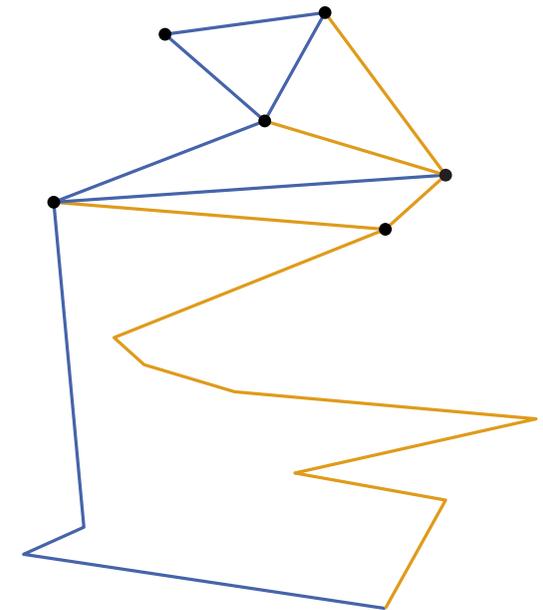
Blatt 2 – Aufgabe 2

Trianguliere y-monotones Polygon

- sortiere Punkte links und rechts nach y-Koordinate

Was funktioniert?

- verbinde im Zickzack alle Ecken von oben nach unten abwechselnd miteinander
- verbinde immer die zwei benachbarten Punkte vom höchsten Punkt
- immer Ohr um aktuellen Punkt abschneiden
- in Komponenten aufteilen und Komponenten triangulieren
- Punkte von oben nach unten abarbeiten, greedy Diagonale
 - auf gleicher Seite: Diagonale falls passende Kurve
 - unterschiedliche Seiten: alle Diagonalen einfügen



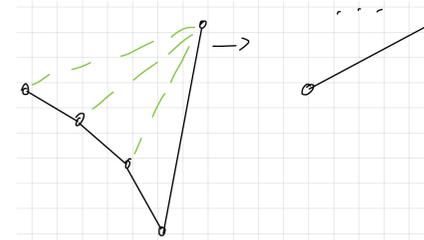
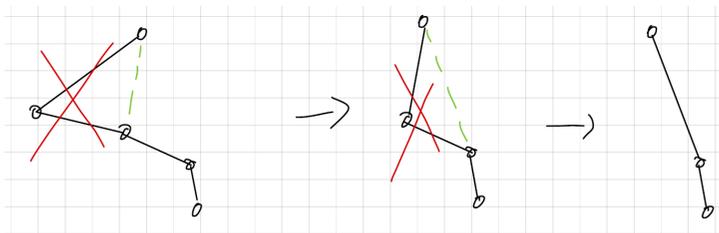
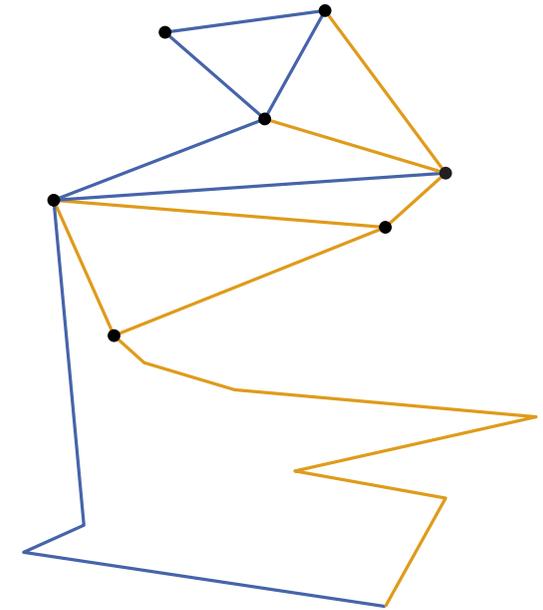
Blatt 2 – Aufgabe 2

Trianguliere y-monotones Polygon

- sortiere Punkte links und rechts nach y-Koordinate

Was funktioniert?

- verbinde im Zickzack alle Ecken von oben nach unten abwechselnd miteinander
- verbinde immer die zwei benachbarten Punkte vom höchsten Punkt
- immer Ohr um aktuellen Punkt abschneiden
- in Komponenten aufteilen und Komponenten triangulieren
- Punkte von oben nach unten abarbeiten, greedy Diagonale
 - auf gleicher Seite: Diagonale falls passende Kurve
 - unterschiedliche Seiten: alle Diagonalen einfügen



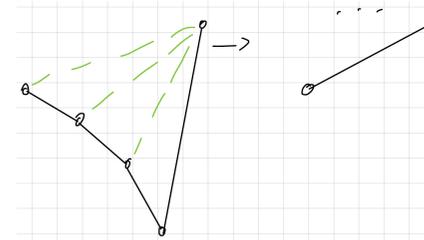
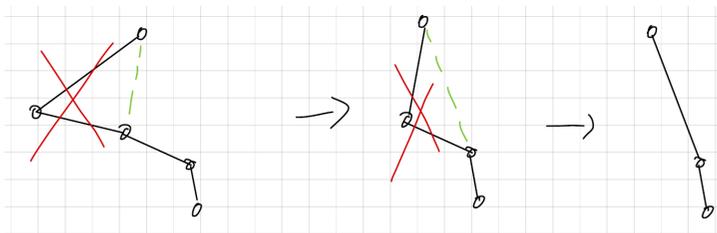
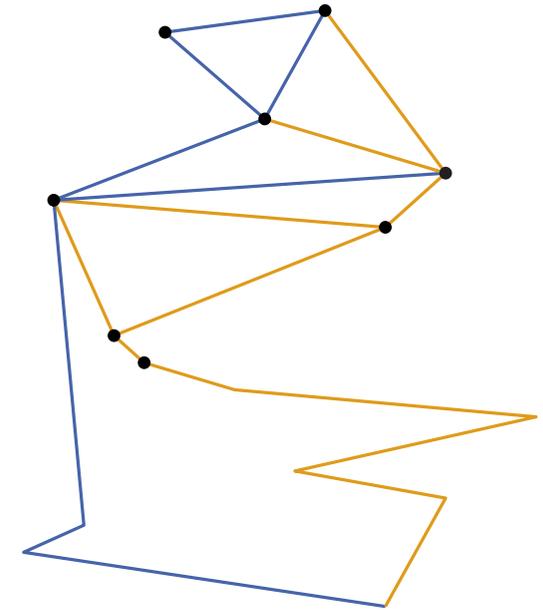
Blatt 2 – Aufgabe 2

Trianguliere y-monotones Polygon

- sortiere Punkte links und rechts nach y-Koordinate

Was funktioniert?

- verbinde im Zickzack alle Ecken von oben nach unten abwechselnd miteinander
- verbinde immer die zwei benachbarten Punkte vom höchsten Punkt
- immer Ohr um aktuellen Punkt abschneiden
- in Komponenten aufteilen und Komponenten triangulieren
- Punkte von oben nach unten abarbeiten, greedy Diagonale
 - auf gleicher Seite: Diagonale falls passende Kurve
 - unterschiedliche Seiten: alle Diagonalen einfügen



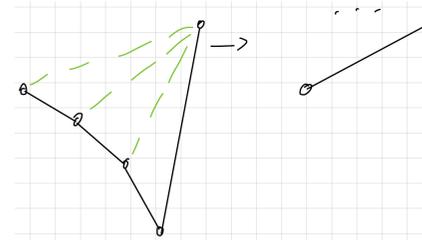
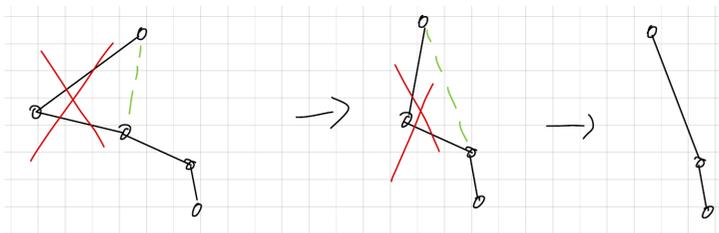
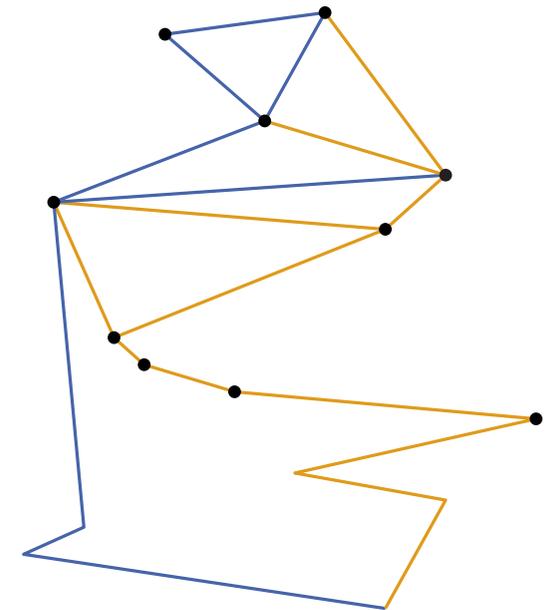
Blatt 2 – Aufgabe 2

Trianguliere y-monotones Polygon

- sortiere Punkte links und rechts nach y-Koordinate

Was funktioniert?

- verbinde im Zickzack alle Ecken von oben nach unten abwechselnd miteinander
- verbinde immer die zwei benachbarten Punkte vom höchsten Punkt
- immer Ohr um aktuellen Punkt abschneiden
- in Komponenten aufteilen und Komponenten triangulieren
- Punkte von oben nach unten abarbeiten, greedy Diagonale
 - auf gleicher Seite: Diagonale falls passende Kurve
 - unterschiedliche Seiten: alle Diagonalen einfügen



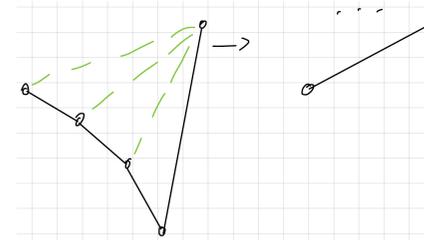
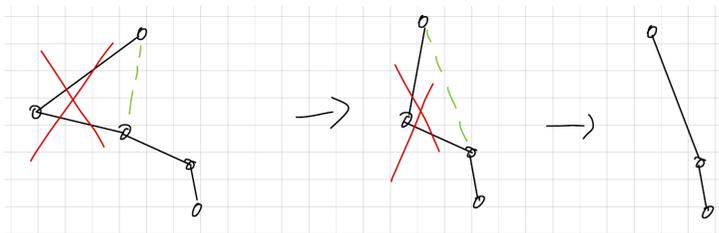
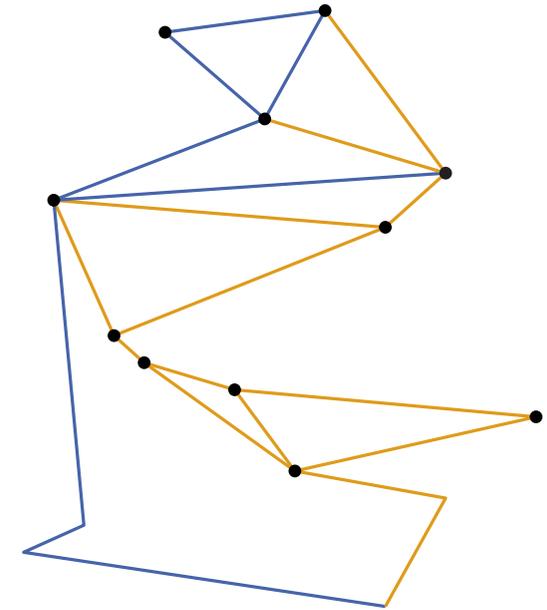
Blatt 2 – Aufgabe 2

Trianguliere y-monotones Polygon

- sortiere Punkte links und rechts nach y-Koordinate

Was funktioniert?

- verbinde im Zickzack alle Ecken von oben nach unten abwechselnd miteinander
- verbinde immer die zwei benachbarten Punkte vom höchsten Punkt
- immer Ohr um aktuellen Punkt abschneiden
- in Komponenten aufteilen und Komponenten triangulieren
- Punkte von oben nach unten abarbeiten, greedy Diagonale
 - auf gleicher Seite: Diagonale falls passende Kurve
 - unterschiedliche Seiten: alle Diagonalen einfügen



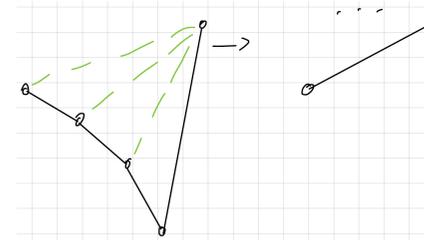
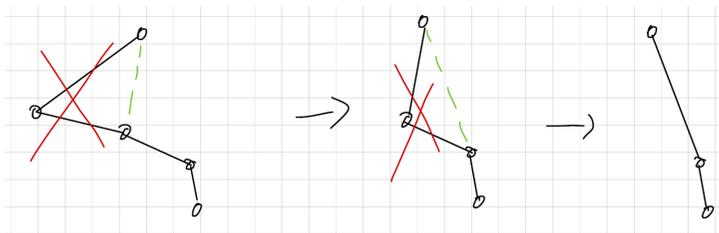
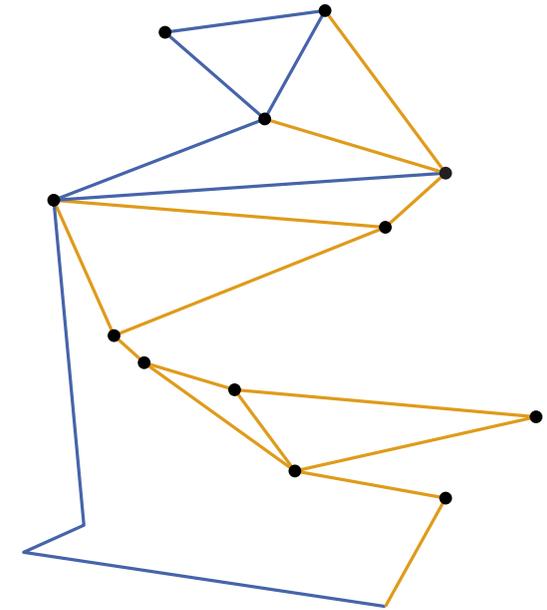
Blatt 2 – Aufgabe 2

Trianguliere y-monotones Polygon

- sortiere Punkte links und rechts nach y-Koordinate

Was funktioniert?

- verbinde im Zickzack alle Ecken von oben nach unten abwechselnd miteinander
- verbinde immer die zwei benachbarten Punkte vom höchsten Punkt
- immer Ohr um aktuellen Punkt abschneiden
- in Komponenten aufteilen und Komponenten triangulieren
- Punkte von oben nach unten abarbeiten, greedy Diagonale
 - auf gleicher Seite: Diagonale falls passende Kurve
 - unterschiedliche Seiten: alle Diagonalen einfügen



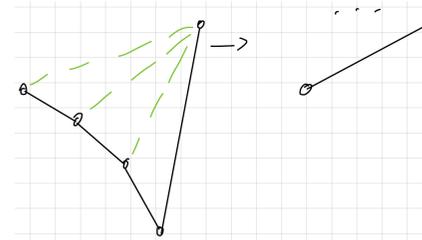
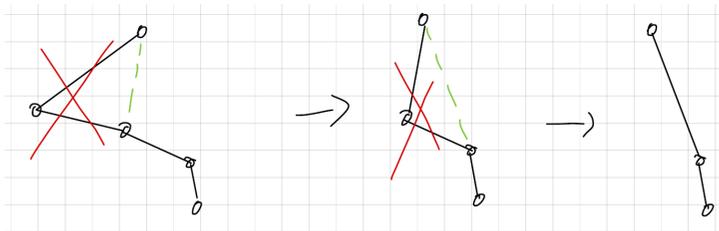
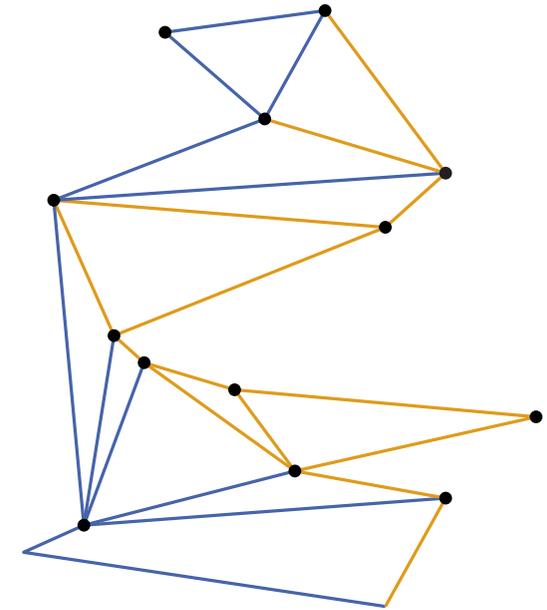
Blatt 2 – Aufgabe 2

Trianguliere y-monotones Polygon

- sortiere Punkte links und rechts nach y-Koordinate

Was funktioniert?

- verbinde im Zickzack alle Ecken von oben nach unten abwechselnd miteinander
- verbinde immer die zwei benachbarten Punkte vom höchsten Punkt
- immer Ohr um aktuellen Punkt abschneiden
- in Komponenten aufteilen und Komponenten triangulieren
- Punkte von oben nach unten abarbeiten, greedy Diagonale
 - auf gleicher Seite: Diagonale falls passende Kurve
 - unterschiedliche Seiten: alle Diagonalen einfügen



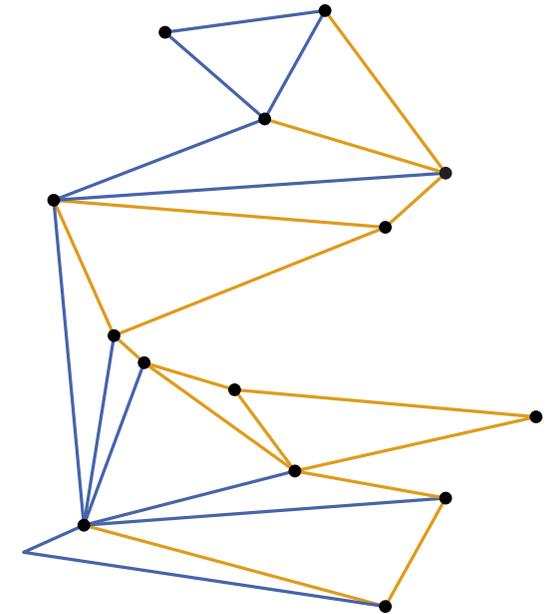
Blatt 2 – Aufgabe 2

Trianguliere y-monotones Polygon

- sortiere Punkte links und rechts nach y-Koordinate

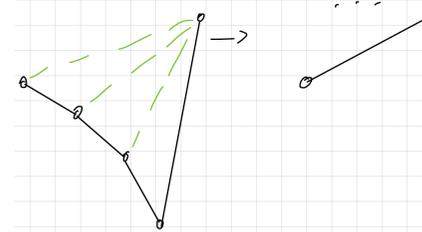
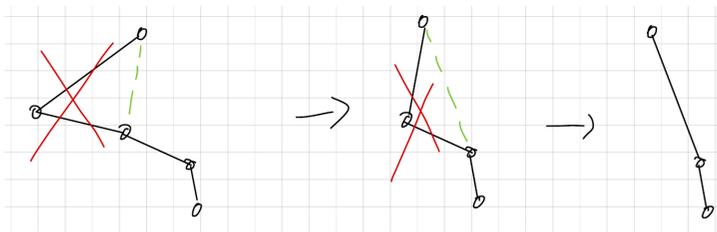
Was funktioniert?

- verbinde im Zickzack alle Ecken von oben nach unten abwechselnd miteinander
- verbinde immer die zwei benachbarten Punkte vom höchsten Punkt
- immer Ohr um aktuellen Punkt abschneiden
- in Komponenten aufteilen und Komponenten triangulieren
- Punkte von oben nach unten abarbeiten, greedy Diagonale
 - auf gleicher Seite: Diagonale falls passende Kurve
 - unterschiedliche Seiten: alle Diagonalen einfügen



Invariante?

Datenstruktur'



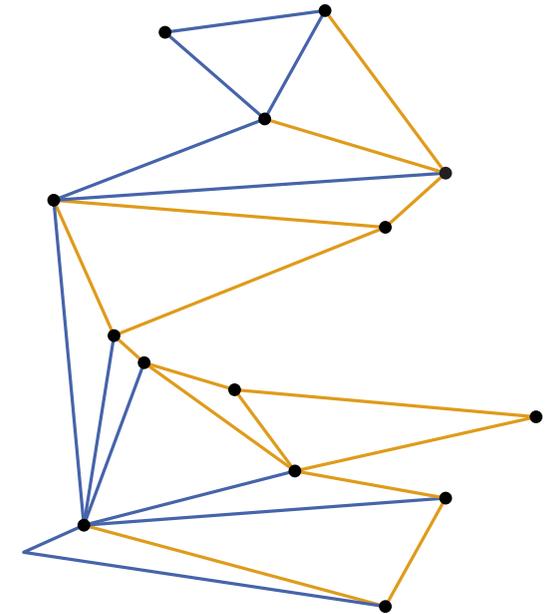
Blatt 2 – Aufgabe 2

Trianguliere y-monotones Polygon

- sortiere Punkte links und rechts nach y-Koordinate

Was funktioniert?

- verbinde im Zickzack alle Ecken von oben nach unten abwechselnd miteinander
- verbinde immer die zwei benachbarten Punkte vom höchsten Punkt
- immer Ohr um aktuellen Punkt abschneiden
- in Komponenten aufteilen und Komponenten triangulieren
- Punkte von oben nach unten abarbeiten, greedy Diagonale
 - auf gleicher Seite: Diagonale falls passende Kurve
 - unterschiedliche Seiten: alle Diagonalen einfügen
- zuerst (1) auf beiden Seiten unabhängig voneinander, dann (2) für beide zusammen

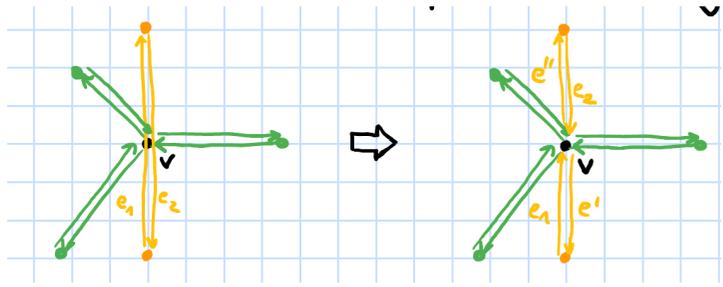
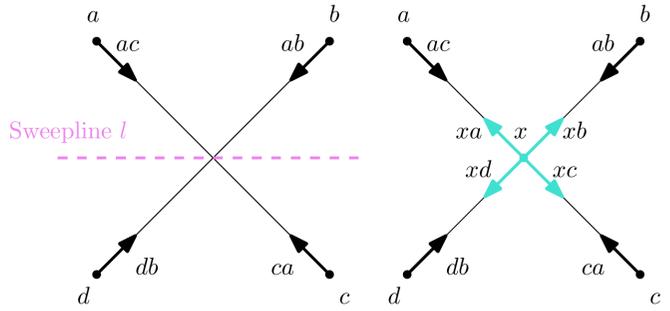


Blatt 2 – Aufgabe 3 + 4

- Schnittpunkt-Sweepeline aus VL

Blatt 2 – Aufgabe 3 + 4

- Schnittpunkt-Sweepeline aus VL
- frickelige Updates der Pointer bei SP

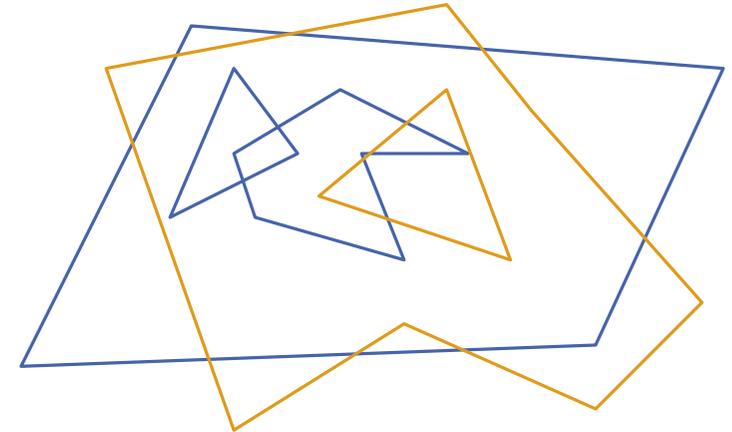


Blatt 2 – Aufgabe 3 + 4

- Schnittpunkt-Sweepeline aus VL
- frickelige Updates der Pointer bei SP

Kinder/Eltern bestimmen

- Zusammenhangskomponenten bestimmen

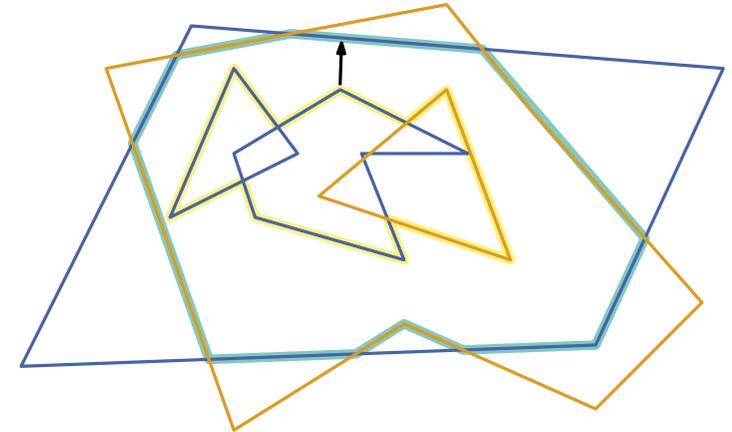


Blatt 2 – Aufgabe 3 + 4

- Schnittpunkt-Sweepeline aus VL
- frickelige Updates der Pointer bei SP

Kinder/Eltern bestimmen

- Zusammenhangskomponenten bestimmen
- in welcher Facette liegt die Komponente?



Blatt 2 – Aufgabe 3 + 4

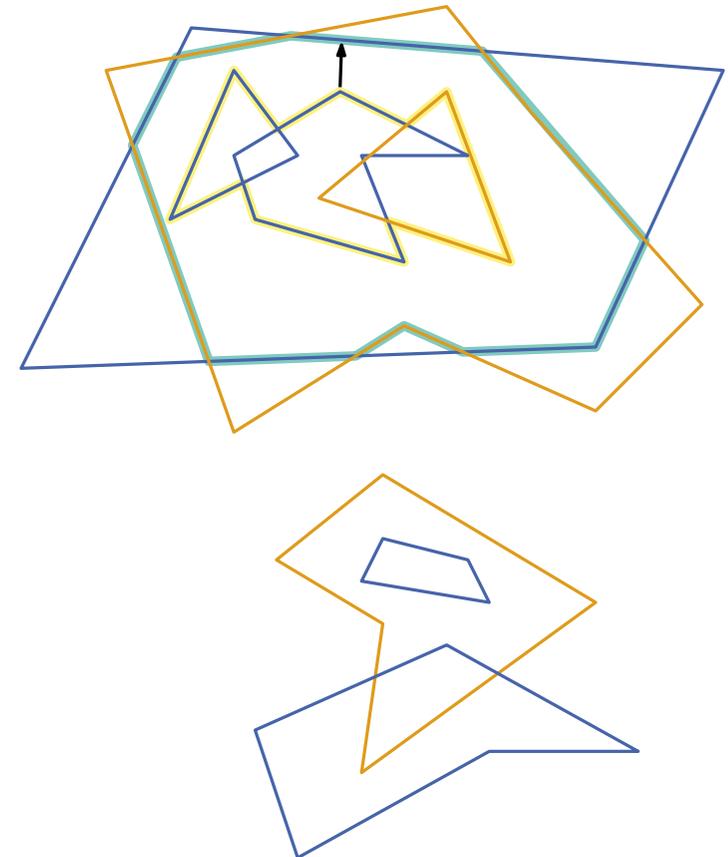
- Schnittpunkt-Sweepeline aus VL
- frickelige Updates der Pointer bei SP

Kinder/Eltern bestimmen

- Zusammenhangskomponenten bestimmen
- in welcher Facette liegt die Komponente?

Label bestimmen

- Facette hat Knoten, der SP ist: einfach



Blatt 2 – Aufgabe 3 + 4

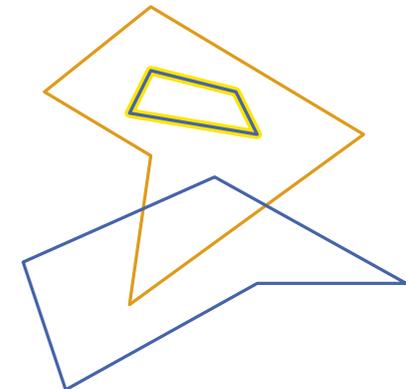
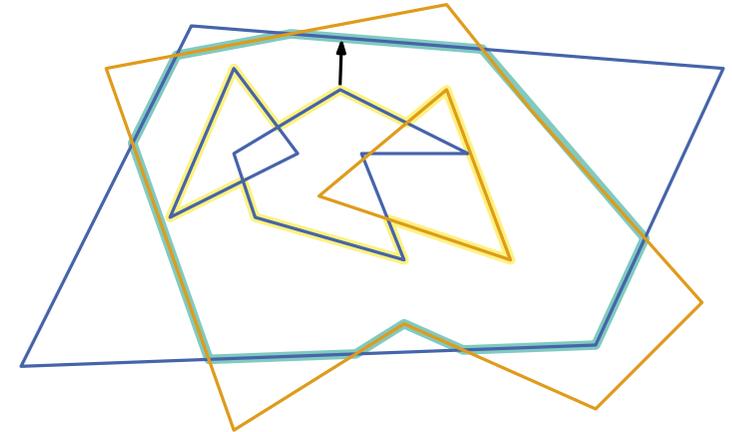
- Schnittpunkt-Sweepeline aus VL
- frickelige Updates der Pointer bei SP

Kinder/Eltern bestimmen

- Zusammenhangskomponenten bestimmen
- in welcher Facette liegt die Komponente?

Label bestimmen

- Facette hat Knoten, der SP ist: einfach
- kein SP: in welcher Facette von G_2 liegt die Facette?



Blatt 2 – Aufgabe 3 + 4

- Schnittpunkt-Sweepeline aus VL
- frickelige Updates der Pointer bei SP

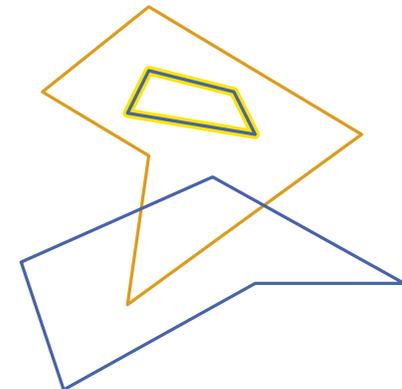
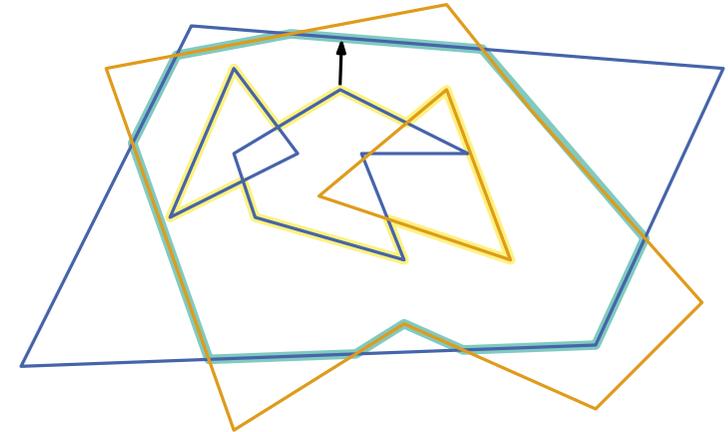
Kinder/Eltern bestimmen

- Zusammenhangskomponenten bestimmen
- in welcher Facette liegt die Komponente?

Label bestimmen

- Facette hat Knoten, der SP ist: einfach
- kein SP: in welcher Facette von G_2 liegt die Facette?

Gegeben: geometrischer Graph G , Punkt P
 In welcher Facette von G liegt P ?



Range Trees

Problem: Range Reporting Query

Gegeben: Punktemenge $P \subseteq \mathbb{R}^d$, Box $B = [a_1, b_1] \times \dots \times [a_d, b_d]$

Welche Punkte sind in B ?

Range Trees

Problem: Range *Counting* Query

Gegeben: Punktemenge $P \subseteq \mathbb{R}^d$, Box $B = [a_1, b_1] \times \dots \times [a_d, b_d]$

Wie viele Punkte aus P sind in B ?

- Wie geht das für $d = 1$? Laufzeit, Speicher?

Range Trees

Problem: Range *Counting* Query

Gegeben: Punktemenge $P \subseteq \mathbb{R}^d$, Box $B = [a_1, b_1] \times \dots \times [a_d, b_d]$

Wie viele Punkte aus P sind in B ?

- Wie geht das für $d = 1$? Laufzeit, Speicher?
- Verallgemeinerung für beliebiges d ?
Vergleiche mit Range Reporting Query!

Range Trees

Problem: Range *Counting* Query

Gegeben: Punktemenge $P \subseteq \mathbb{R}^d$, Box $B = [a_1, b_1] \times \dots \times [a_d, b_d]$

Wie viele Punkte aus P sind in B ?

- Wie geht das für $d = 1$? Laufzeit, Speicher?
- Verallgemeinerung für beliebiges d ?
Vergleiche mit Range Reporting Query!
- Geht das für $d = 2$ besser?

Range Trees

Problem: Range *Counting* Query

Gegeben: Punktemenge $P \subseteq \mathbb{R}^d$, Box $B = [a_1, b_1] \times \dots \times [a_d, b_d]$

Wie viele Punkte aus P sind in B ?

- Wie geht das für $d = 1$? Laufzeit, Speicher?
- Verallgemeinerung für beliebiges d ?
Vergleiche mit Range Reporting Query!
- Geht das für $d = 2$ besser?
- Geht das für $d = 3$ besser?

Range Trees

Problem: Range *Counting* Query

Gegeben: Punktemenge $P \subseteq \mathbb{R}^d$, Box $B = [a_1, b_1] \times \dots \times [a_d, b_d]$

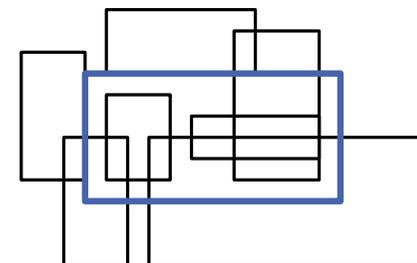
Wie viele Punkte aus P sind in B ?

- Wie geht das für $d = 1$? Laufzeit, Speicher?
- Verallgemeinerung für beliebiges d ?
Vergleiche mit Range Reporting Query!
- Geht das für $d = 2$ besser?
- Geht das für $d = 3$ besser?

Problem: Range *Rectangle Reporting* Query

Gegeben: achsenparallele Rechtecke R , Box $B = [a_1, b_1] \times [a_2, b_2]$

Welche Rechtecke R sind vollständig in B ?



Range Trees

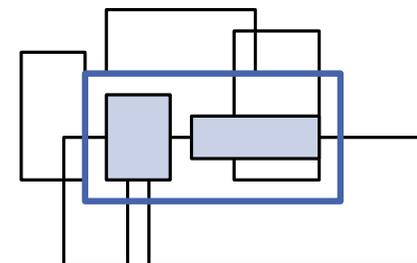
Problem: Range *Counting* Query

Gegeben: Punktemenge $P \subseteq \mathbb{R}^d$, Box $B = [a_1, b_1] \times \dots \times [a_d, b_d]$
Wie viele Punkte aus P sind in B ?

- Wie geht das für $d = 1$? Laufzeit, Speicher?
- Verallgemeinerung für beliebiges d ?
 Vergleiche mit Range Reporting Query!
- Geht das für $d = 2$ besser?
- Geht das für $d = 3$ besser?

Problem: Range *Rectangle Reporting* Query

Gegeben: achsenparallele Rechtecke R , Box $B = [a_1, b_1] \times [a_2, b_2]$
Welche Rechtecke R sind vollständig in B ?



Range Trees

Problem: Range *Counting* Query

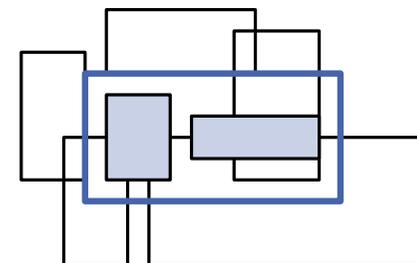
Gegeben: Punktemenge $P \subseteq \mathbb{R}^d$, Box $B = [a_1, b_1] \times \dots \times [a_d, b_d]$
Wie viele Punkte aus P sind in B ?

- Wie geht das für $d = 1$? Laufzeit, Speicher?
- Verallgemeinerung für beliebiges d ?
 Vergleiche mit Range Reporting Query!
- Geht das für $d = 2$ besser?
- Geht das für $d = 3$ besser?

Problem: Range *Rectangle* Reporting Query

Gegeben: achsenparallele Rechtecke R , Box $B = [a_1, b_1] \times [a_2, b_2]$
 Welche Rechtecke R sind vollständig in B ?

- Welche Datenstruktur? Aufbau? Laufzeit? Speicher?
Tipp: ubpu-qvzrafvbany



Range Trees

Problem: Range *Counting* Query

Gegeben: Punktemenge $P \subseteq \mathbb{R}^d$, Box $B = [a_1, b_1] \times \dots \times [a_d, b_d]$
Wie viele Punkte aus P sind in B ?

- Wie geht das für $d = 1$? Laufzeit, Speicher?
- Verallgemeinerung für beliebiges d ?
 Vergleiche mit Range Reporting Query!
- Geht das für $d = 2$ besser?
- Geht das für $d = 3$ besser?

Problem: Range *Rectangle* Reporting Query

Gegeben: achsenparallele Rechtecke R , Box $B = [a_1, b_1] \times [a_2, b_2]$
Welche Rechtecke R sind vollständig in B ?

- Welche Datenstruktur? Aufbau? Laufzeit? Speicher?
Tipp: ubpu-qvzrafvbany
- beliebige Polygone statt Rechtecke?

