

Algorithmische Geometrie

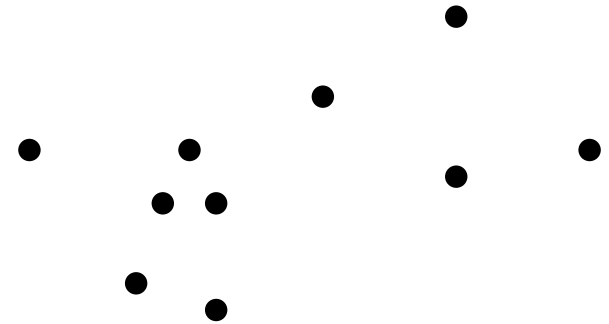
Aktivsession 1



Konvexe Hülle von Polygonen

Graham Scan

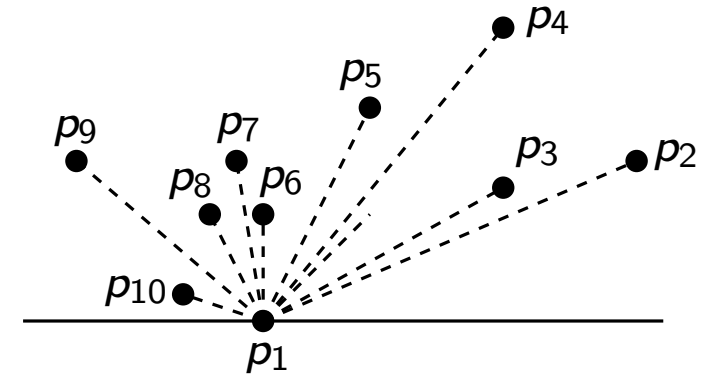
- starte bei unterstem Punkt
- sortiere nach Winkel zur Horizontalen
- sonst: wie Andrews Algorithmus



Konvexe Hülle von Polygonen

Graham Scan

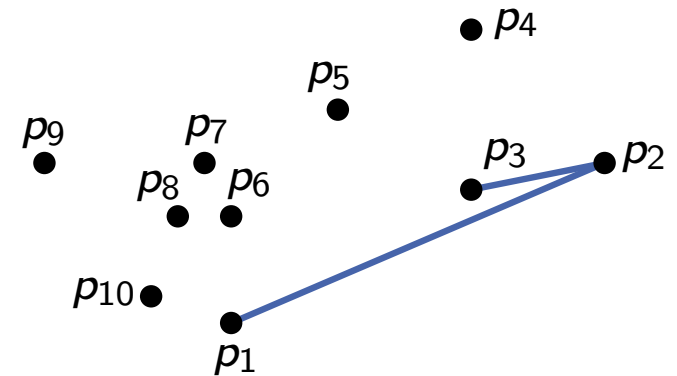
- starte bei unterstem Punkt
- sortiere nach Winkel zur Horizontalen
- sonst: wie Andrews Algorithmus



Konvexe Hülle von Polygonen

Graham Scan

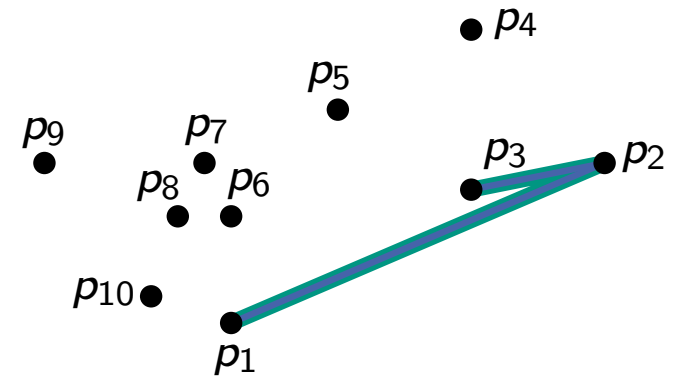
- starte bei unterstem Punkt
- sortiere nach Winkel zur Horizontalen
- sonst: wie Andrews Algorithmus



Konvexe Hülle von Polygonen

Graham Scan

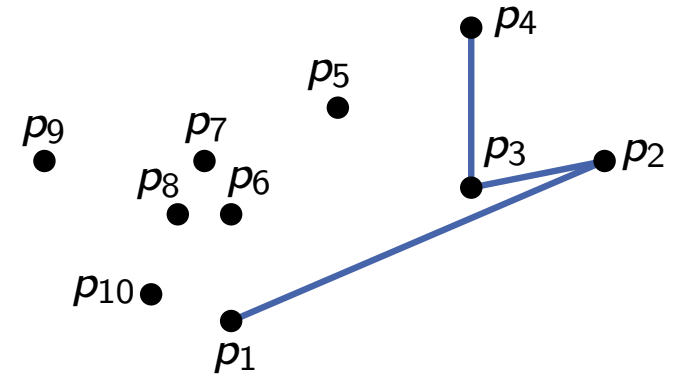
- starte bei unterstem Punkt
- sortiere nach Winkel zur Horizontalen
- sonst: wie Andrews Algorithmus



Konvexe Hülle von Polygonen

Graham Scan

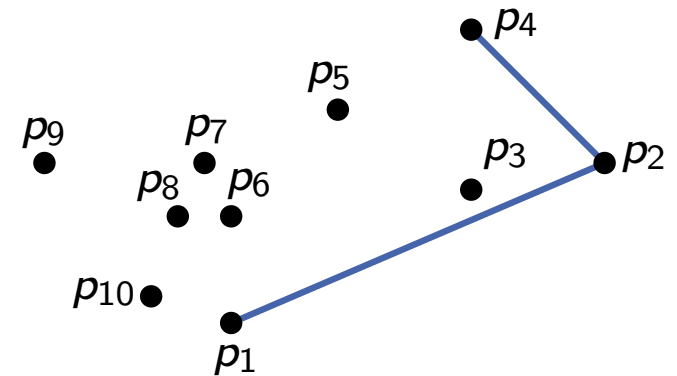
- starte bei unterstem Punkt
- sortiere nach Winkel zur Horizontalen
- sonst: wie Andrews Algorithmus



Konvexe Hülle von Polygonen

Graham Scan

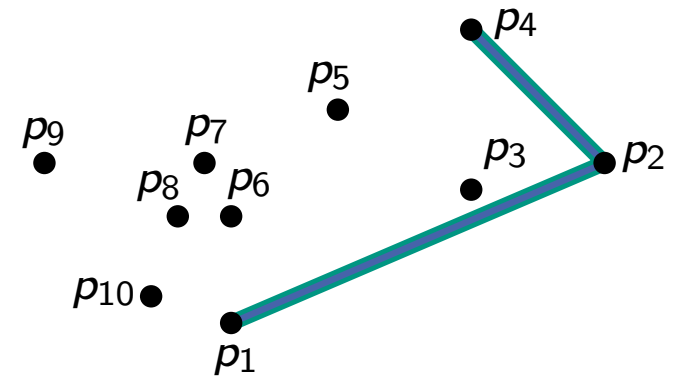
- starte bei unterstem Punkt
- sortiere nach Winkel zur Horizontalen
- sonst: wie Andrews Algorithmus



Konvexe Hülle von Polygonen

Graham Scan

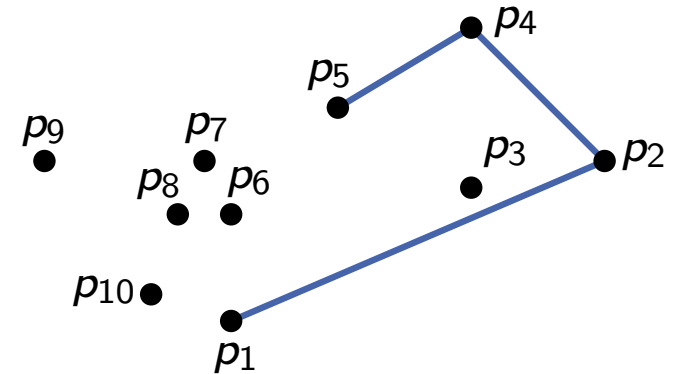
- starte bei unterstem Punkt
- sortiere nach Winkel zur Horizontalen
- sonst: wie Andrews Algorithmus



Konvexe Hülle von Polygonen

Graham Scan

- starte bei unterstem Punkt
- sortiere nach Winkel zur Horizontalen
- sonst: wie Andrews Algorithmus



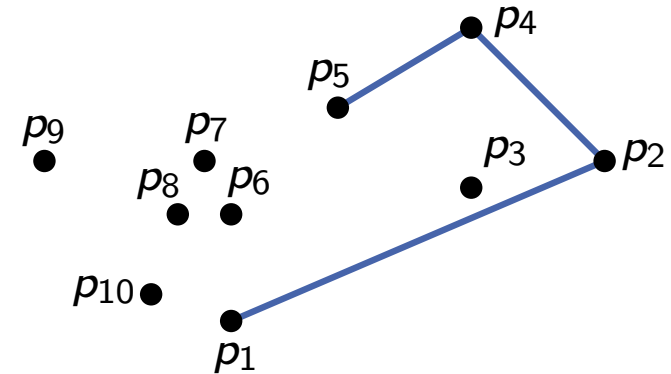
Konvexe Hülle von Polygonen

Graham Scan

- starte bei unterstem Punkt
- sortiere nach Winkel zur Horizontalen
- sonst: wie Andrews Algorithmus

Korrektheit

- das Polygon ist zu jedem Zeitpunkt konvex
- gelöschte Punkte sind nicht Eckpunkt der konvexen Hülle



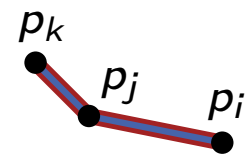
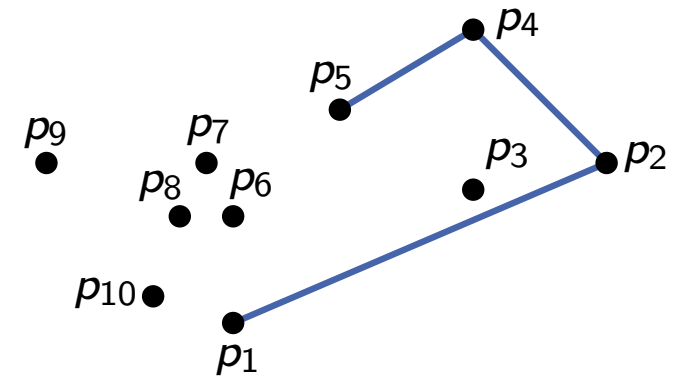
Konvexe Hülle von Polygonen

Graham Scan

- starte bei unterstem Punkt
- sortiere nach Winkel zur Horizontalen
- sonst: wie Andrews Algorithmus

Korrektheit

- das Polygon ist zu jedem Zeitpunkt konvex
- gelöschte Punkte sind nicht Eckpunkt der konvexen Hülle
 - betrachte entfernten Punkt p_j , Nachfolger p_k , Vorgänger p_i



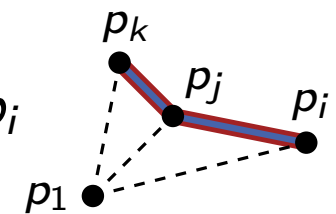
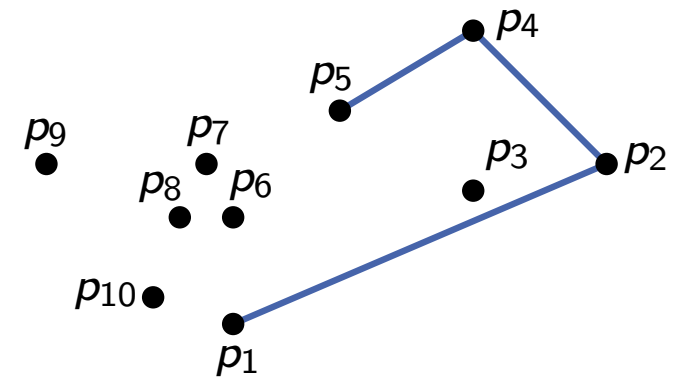
Konvexe Hülle von Polygonen

Graham Scan

- starte bei unterstem Punkt
- sortiere nach Winkel zur Horizontalen
- sonst: wie Andrews Algorithmus

Korrektheit

- das Polygon ist zu jedem Zeitpunkt konvex
- gelöschte Punkte sind nicht Eckpunkt der konvexen Hülle
 - betrachte entfernten Punkt p_j , Nachfolger p_k , Vorgänger p_i
 - p_j liegt im Inneren des Dreiecks $p_i p_k p_1$



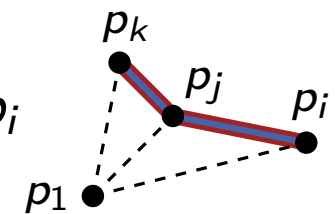
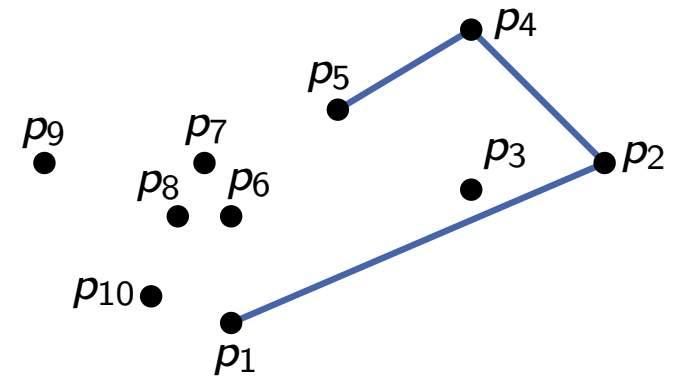
Konvexe Hülle von Polygonen

Graham Scan

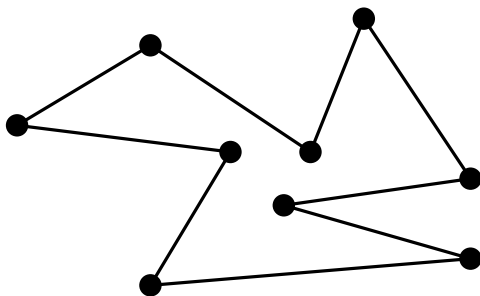
- starte bei unterstem Punkt
- sortiere nach Winkel zur Horizontalen
- sonst: wie Andrews Algorithmus

Korrektheit

- das Polygon ist zu jedem Zeitpunkt konvex
- gelöschte Punkte sind nicht Eckpunkt der konvexen Hülle
 - betrachte entfernten Punkt p_j , Nachfolger p_k , Vorgänger p_i
 - p_j liegt im Inneren des Dreiecks $p_i p_k p_1$



Eingabe: einfaches Polygon



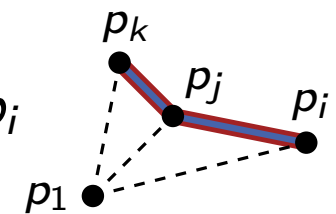
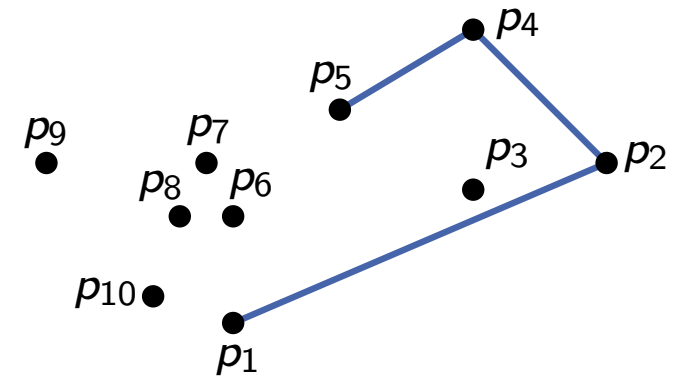
Konvexe Hülle von Polygonen

Graham Scan

- starte bei unterstem Punkt
- sortiere nach Winkel zur Horizontalen
- sonst: wie Andrews Algorithmus

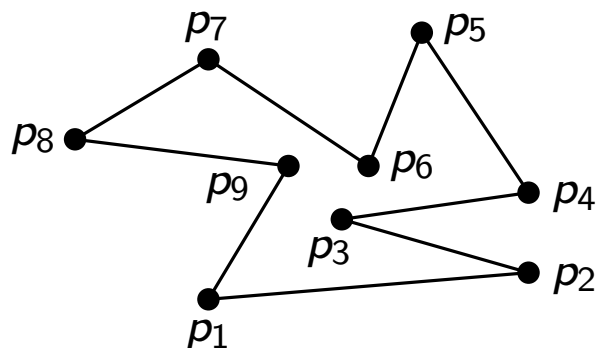
Korrektheit

- das Polygon ist zu jedem Zeitpunkt konvex
- gelöschte Punkte sind nicht Eckpunkt der konvexen Hülle
 - betrachte entfernten Punkt p_j , Nachfolger p_k , Vorgänger p_i
 - p_j liegt im Inneren des Dreiecks $p_i p_k p_1$



Eingabe: einfaches Polygon

- füge Punkte entsprechend Reihenfolge im Polygon ein



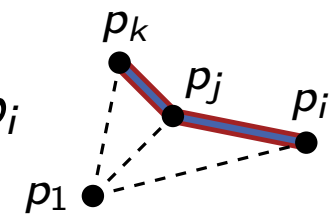
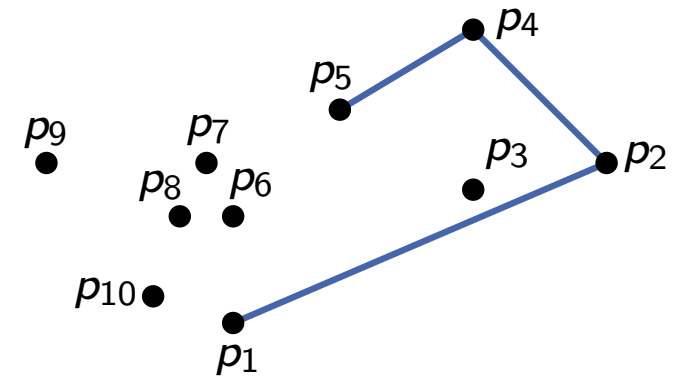
Konvexe Hülle von Polygonen

Graham Scan

- starte bei unterstem Punkt
- sortiere nach Winkel zur Horizontalen
- sonst: wie Andrews Algorithmus

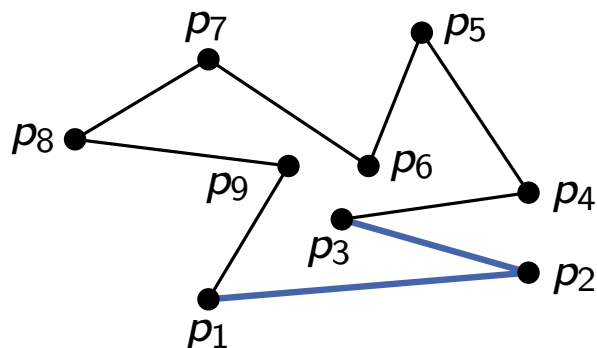
Korrektheit

- das Polygon ist zu jedem Zeitpunkt konvex
- gelöschte Punkte sind nicht Eckpunkt der konvexen Hülle
 - betrachte entfernten Punkt p_j , Nachfolger p_k , Vorgänger p_i
 - p_j liegt im Inneren des Dreiecks $p_i p_k p_1$



Eingabe: einfaches Polygon

- füge Punkte entsprechend Reihenfolge im Polygon ein



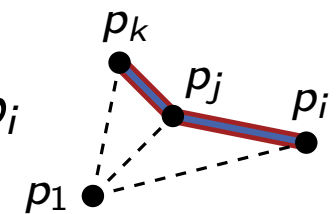
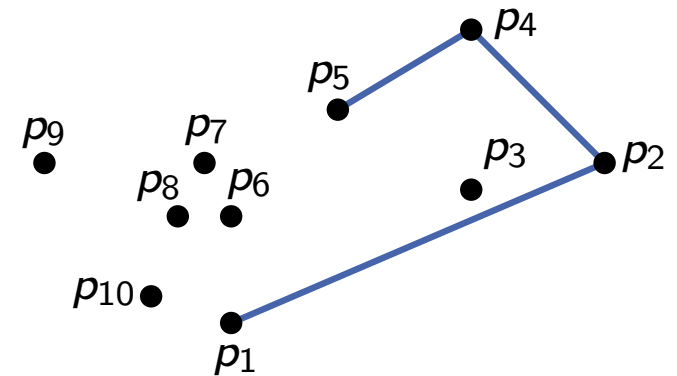
Konvexe Hülle von Polygonen

Graham Scan

- starte bei unterstem Punkt
- sortiere nach Winkel zur Horizontalen
- sonst: wie Andrews Algorithmus

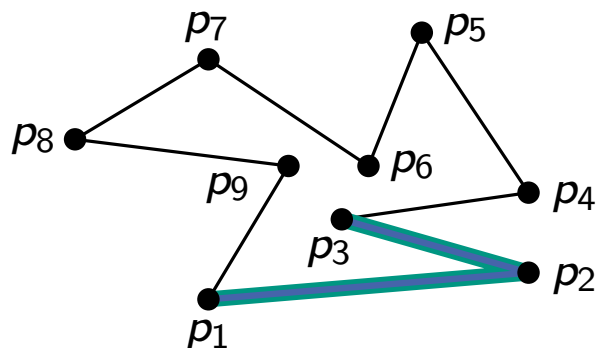
Korrektheit

- das Polygon ist zu jedem Zeitpunkt konvex
- gelöschte Punkte sind nicht Eckpunkt der konvexen Hülle
 - betrachte entfernten Punkt p_j , Nachfolger p_k , Vorgänger p_i
 - p_j liegt im Inneren des Dreiecks $p_i p_k p_1$



Eingabe: einfaches Polygon

- füge Punkte entsprechend Reihenfolge im Polygon ein



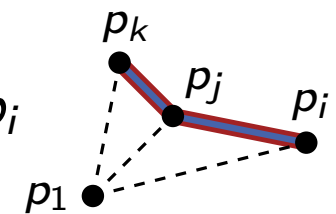
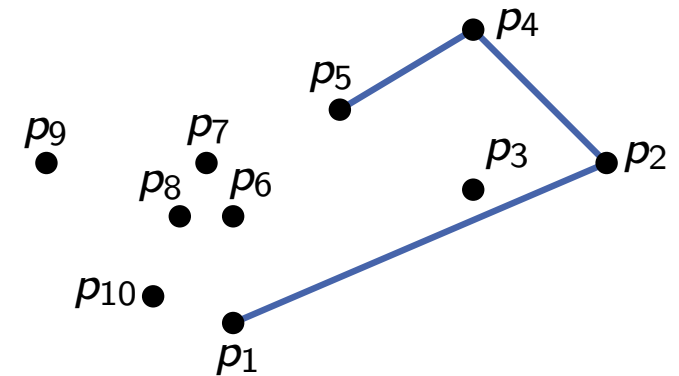
Konvexe Hülle von Polygonen

Graham Scan

- starte bei unterstem Punkt
- sortiere nach Winkel zur Horizontalen
- sonst: wie Andrews Algorithmus

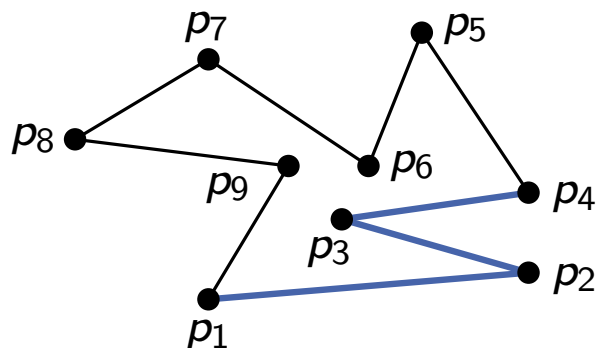
Korrektheit

- das Polygon ist zu jedem Zeitpunkt konvex
- gelöschte Punkte sind nicht Eckpunkt der konvexen Hülle
 - betrachte entfernten Punkt p_j , Nachfolger p_k , Vorgänger p_i
 - p_j liegt im Inneren des Dreiecks $p_i p_k p_1$



Eingabe: einfaches Polygon

- füge Punkte entsprechend Reihenfolge im Polygon ein



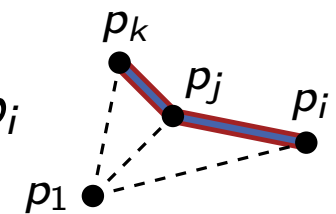
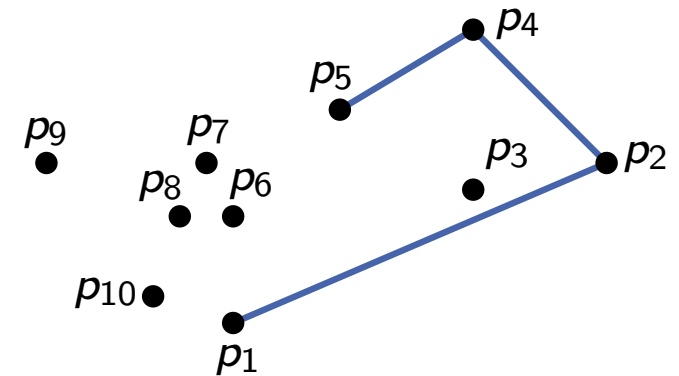
Konvexe Hülle von Polygonen

Graham Scan

- starte bei unterstem Punkt
- sortiere nach Winkel zur Horizontalen
- sonst: wie Andrews Algorithmus

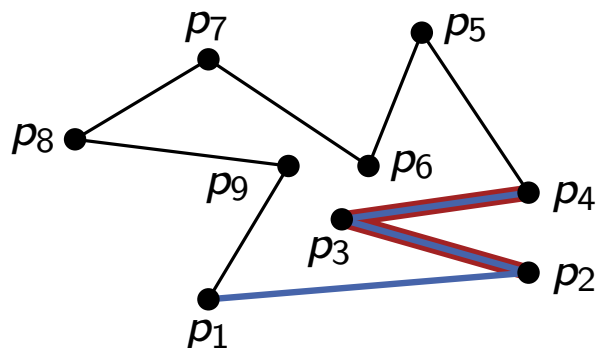
Korrektheit

- das Polygon ist zu jedem Zeitpunkt konvex
- gelöschte Punkte sind nicht Eckpunkt der konvexen Hülle
 - betrachte entfernten Punkt p_j , Nachfolger p_k , Vorgänger p_i
 - p_j liegt im Inneren des Dreiecks $p_i p_k p_1$



Eingabe: einfaches Polygon

- füge Punkte entsprechend Reihenfolge im Polygon ein



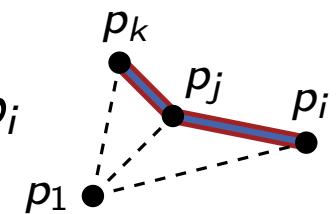
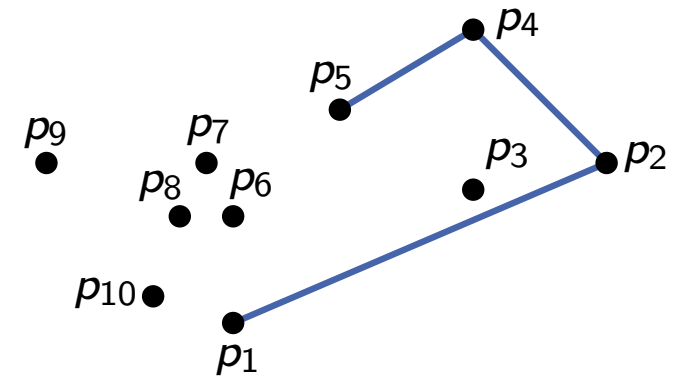
Konvexe Hülle von Polygonen

Graham Scan

- starte bei unterstem Punkt
- sortiere nach Winkel zur Horizontalen
- sonst: wie Andrews Algorithmus

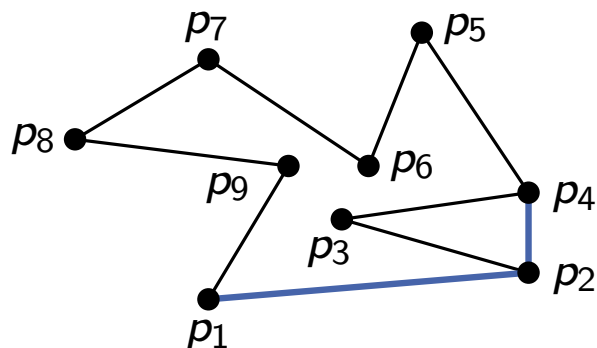
Korrektheit

- das Polygon ist zu jedem Zeitpunkt konvex
- gelöschte Punkte sind nicht Eckpunkt der konvexen Hülle
 - betrachte entfernten Punkt p_j , Nachfolger p_k , Vorgänger p_i
 - p_j liegt im Inneren des Dreiecks $p_i p_k p_1$



Eingabe: einfaches Polygon

- füge Punkte entsprechend Reihenfolge im Polygon ein



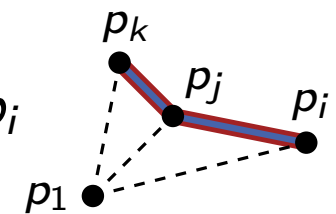
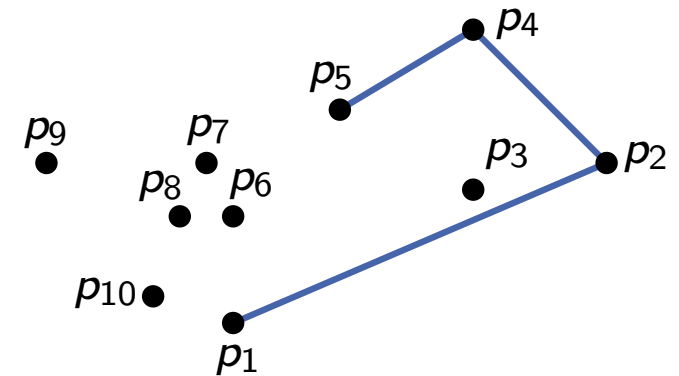
Konvexe Hülle von Polygonen

Graham Scan

- starte bei unterstem Punkt
- sortiere nach Winkel zur Horizontalen
- sonst: wie Andrews Algorithmus

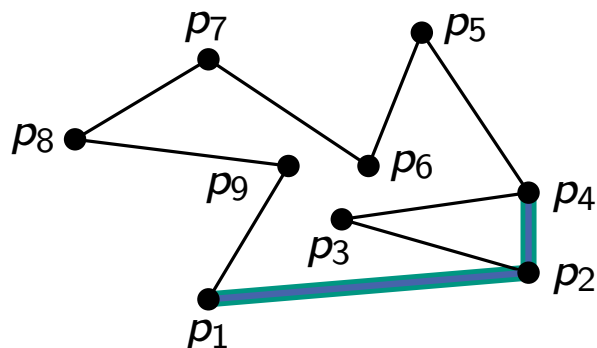
Korrektheit

- das Polygon ist zu jedem Zeitpunkt konvex
- gelöschte Punkte sind nicht Eckpunkt der konvexen Hülle
 - betrachte entfernten Punkt p_j , Nachfolger p_k , Vorgänger p_i
 - p_j liegt im Inneren des Dreiecks $p_i p_k p_1$



Eingabe: einfaches Polygon

- füge Punkte entsprechend Reihenfolge im Polygon ein



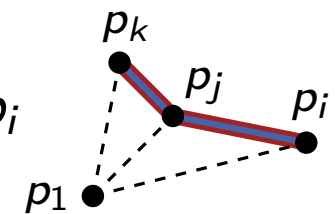
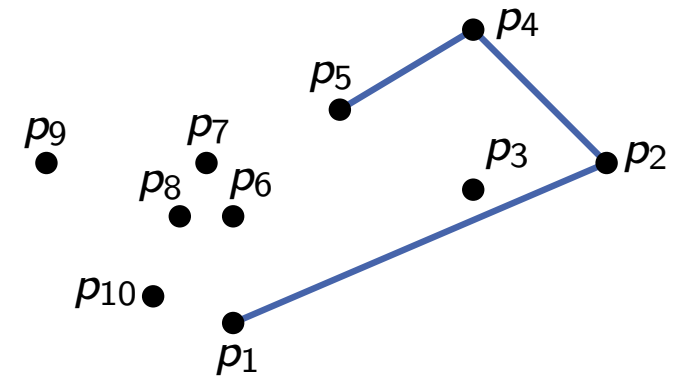
Konvexe Hülle von Polygonen

Graham Scan

- starte bei unterstem Punkt
- sortiere nach Winkel zur Horizontalen
- sonst: wie Andrews Algorithmus

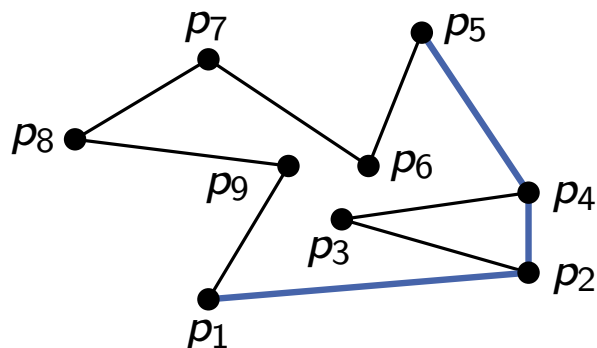
Korrektheit

- das Polygon ist zu jedem Zeitpunkt konvex
- gelöschte Punkte sind nicht Eckpunkt der konvexen Hülle
 - betrachte entfernten Punkt p_j , Nachfolger p_k , Vorgänger p_i
 - p_j liegt im Inneren des Dreiecks $p_i p_k p_1$



Eingabe: einfaches Polygon

- füge Punkte entsprechend Reihenfolge im Polygon ein



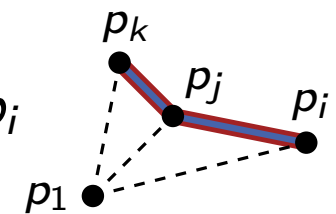
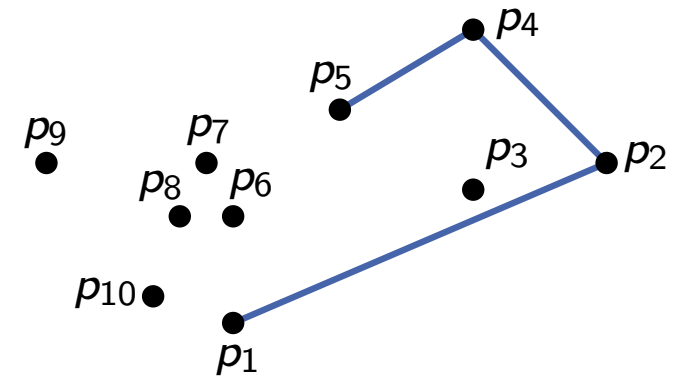
Konvexe Hülle von Polygonen

Graham Scan

- starte bei unterstem Punkt
- sortiere nach Winkel zur Horizontalen
- sonst: wie Andrews Algorithmus

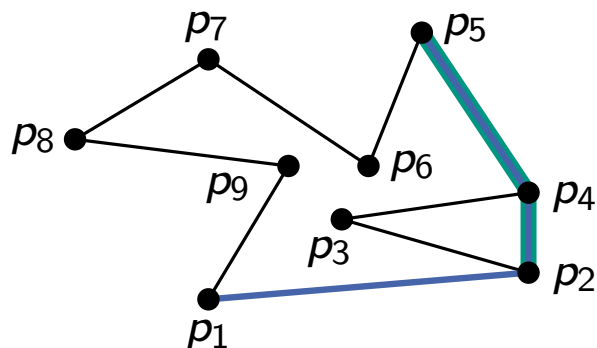
Korrektheit

- das Polygon ist zu jedem Zeitpunkt konvex
- gelöschte Punkte sind nicht Eckpunkt der konvexen Hülle
 - betrachte entfernten Punkt p_j , Nachfolger p_k , Vorgänger p_i
 - p_j liegt im Inneren des Dreiecks $p_i p_k p_1$



Eingabe: einfaches Polygon

- füge Punkte entsprechend Reihenfolge im Polygon ein



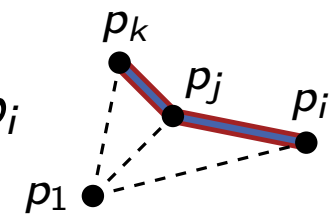
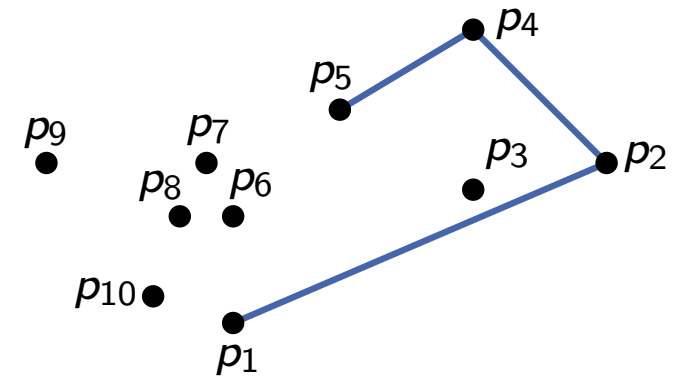
Konvexe Hülle von Polygonen

Graham Scan

- starte bei unterstem Punkt
- sortiere nach Winkel zur Horizontalen
- sonst: wie Andrews Algorithmus

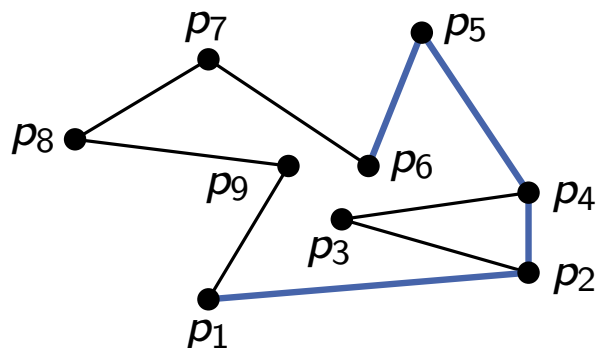
Korrektheit

- das Polygon ist zu jedem Zeitpunkt konvex
- gelöschte Punkte sind nicht Eckpunkt der konvexen Hülle
 - betrachte entfernten Punkt p_j , Nachfolger p_k , Vorgänger p_i
 - p_j liegt im Inneren des Dreiecks $p_i p_k p_1$



Eingabe: einfaches Polygon

- füge Punkte entsprechend Reihenfolge im Polygon ein



Konvexe Hülle von Polygonen

Graham Scan

- starte bei unterstem Punkt
- sortiere nach Winkel zur Horizontalen
- sonst: wie Andrews Algorithmus

Korrektheit

stimmt weiterhin

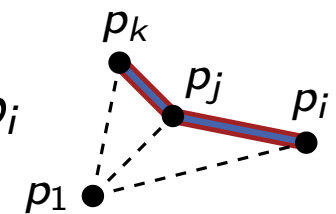
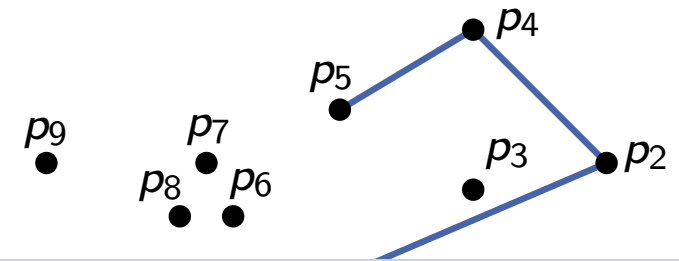
- das Polygon ist zu jedem Zeitpunkt konvex

- gelöschte Punkte sind nicht Eckpunkt der konvexen Hülle

- betrachte entfernten Punkt p_j , Nachfolger p_k , Vorgänger p_i
- p_j liegt im Inneren des Dreiecks $p_i p_k p_1$

stimmt ebenfalls

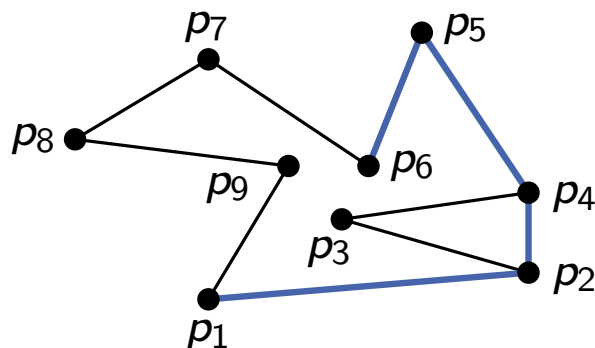
(muss man aber anders zeigen)



Eingabe: einfaches Polygon

- füge Punkte entsprechend Reihenfolge im Polygon ein

Korrektheit



Konvexe Hülle von Polygonen

Graham Scan

- starte bei unterstem Punkt
- sortiere nach Winkel zur Horizontalen
- sonst: wie Andrews Algorithmus

Korrektheit

stimmt weiterhin

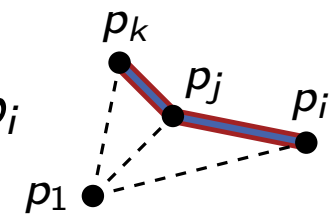
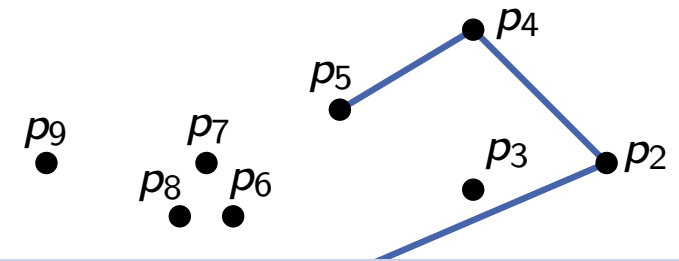
- das Polygon ist zu jedem Zeitpunkt konvex

- gelöschte Punkte sind nicht Eckpunkt der konvexen Hülle

- betrachte entfernten Punkt p_j , Nachfolger p_k , Vorgänger p_i
- p_j liegt im Inneren des Dreiecks $p_i p_k p_1$

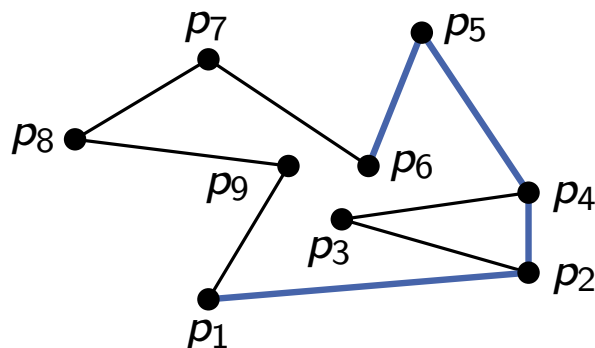
stimmt ebenfalls

(muss man aber anders zeigen)

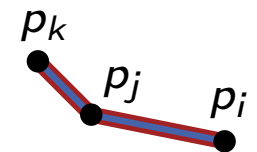


Eingabe: einfaches Polygon

- füge Punkte entsprechend Reihenfolge im Polygon ein



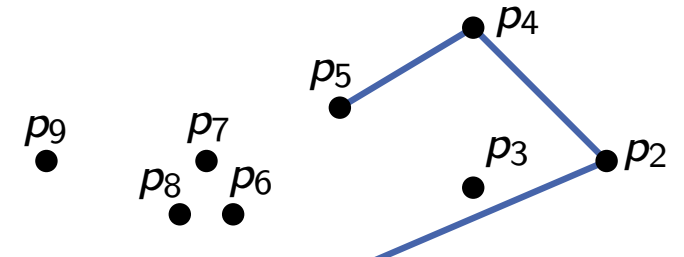
Korrektheit



Konvexe Hülle von Polygonen

Graham Scan

- starte bei unterstem Punkt
- sortiere nach Winkel zur Horizontalen
- sonst: wie Andrews Algorithmus



Korrektheit

stimmt weiterhin

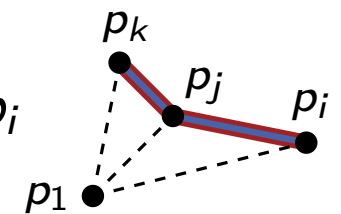
stimmt ebenfalls

(muss man aber anders zeigen)

■ das Polygon ist zu jedem Zeitpunkt konvex

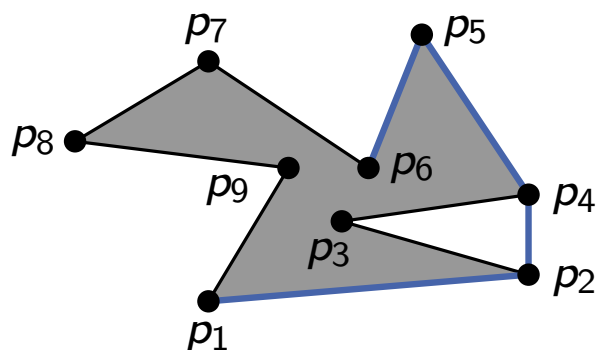
■ gelöschte Punkte sind nicht Eckpunkt der konvexen Hülle

- betrachte entfernten Punkt p_j , Nachfolger p_k , Vorgänger p_i
- p_j liegt im Inneren des Dreiecks $p_i p_k p_1$



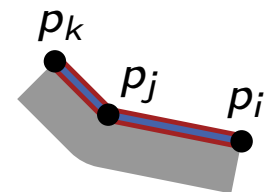
Eingabe: einfaches Polygon

■ füge Punkte entsprechend Reihenfolge im Polygon ein



Korrektheit

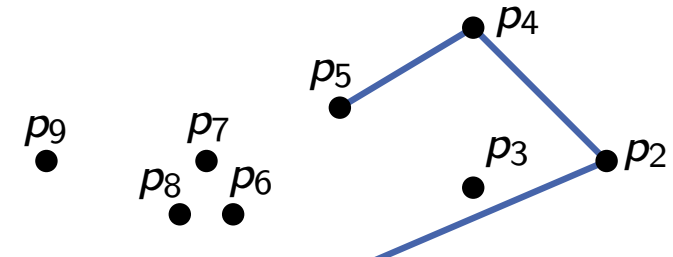
- wir laufen gegen den Uhrzeigersinn
- das Innere des Polygons liegt links



Konvexe Hülle von Polygonen

Graham Scan

- starte bei unterstem Punkt
- sortiere nach Winkel zur Horizontalen
- sonst: wie Andrews Algorithmus



Korrektheit

stimmt weiterhin

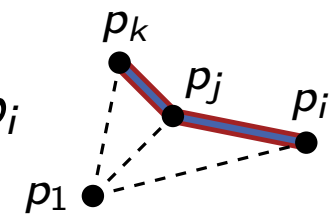
stimmt ebenfalls

(muss man aber anders zeigen)

- das Polygon ist zu jedem Zeitpunkt konvex

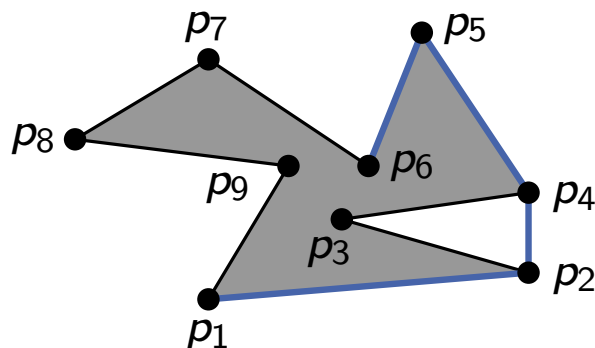
- gelöschte Punkte sind nicht Eckpunkt der konvexen Hülle

- betrachte entfernten Punkt p_j , Nachfolger p_k , Vorgänger p_i
- p_j liegt im Inneren des Dreiecks $p_i p_k p_1$



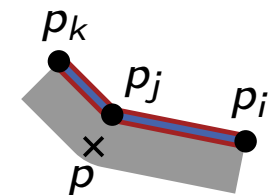
Eingabe: einfaches Polygon

- füge Punkte entsprechend Reihenfolge im Polygon ein



Korrektheit

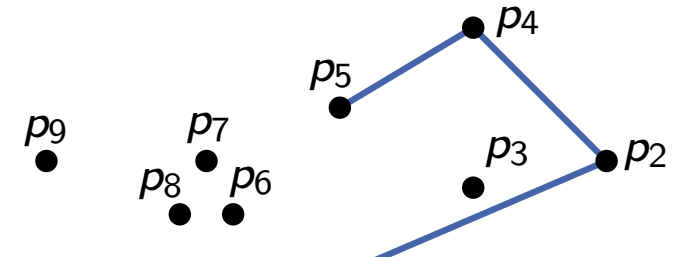
- wir laufen gegen den Uhrzeigersinn
- das Innere des Polygons liegt links
- es gibt Punkt p in der konvexen Hülle, sodass p_j im Dreieck $p_i p_k p$ liegt



Konvexe Hülle von Polygonen

Graham Scan

- starte bei unterstem Punkt
- sortiere nach Winkel zur Horizontalen
- sonst: wie Andrews Algorithmus



Korrektheit

■ das Polygon ist zu jedem Zeitpunkt konvex

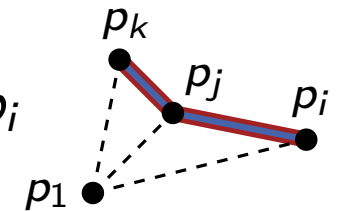
stimmt weiterhin

■ gelöschte Punkte sind nicht Eckpunkt der konvexen Hülle

stimmt ebenfalls

(muss man aber anders zeigen)

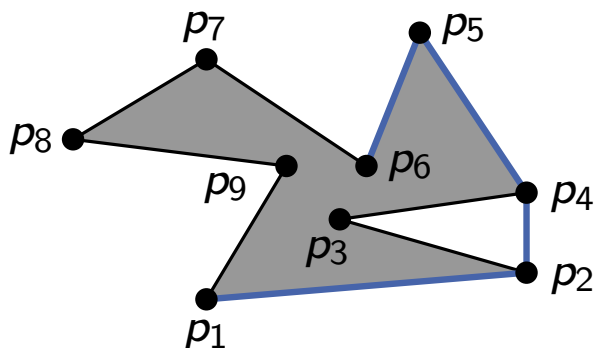
- betrachte entfernten Punkt p_j , Nachfolger p_k , Vorgänger p_i
- p_j liegt im Inneren des Dreiecks $p_i p_k p_1$



Eingabe: einfaches Polygon

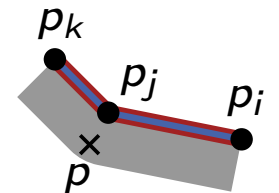
Warum ist der Algorithmus nicht korrekt?

■ füge Punkte entsprechend Reihenfolge im Polygon ein



Korrektheit

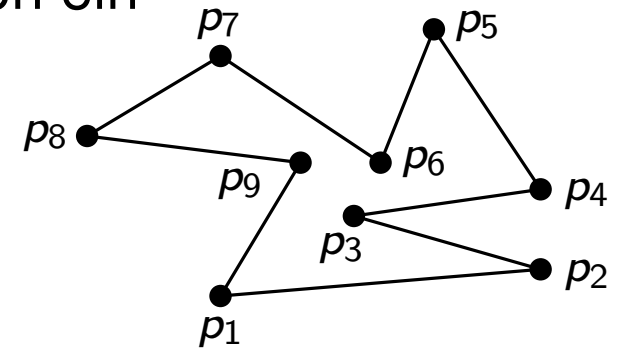
- wir laufen gegen den Uhrzeigersinn
- das Innere des Polygons liegt links
- es gibt Punkt p in der konvexen Hülle, sodass p_j im Dreieck $p_i p_k p$ liegt



Wie geht es richtig?

Ansatz für einen korrekten Algorithmus

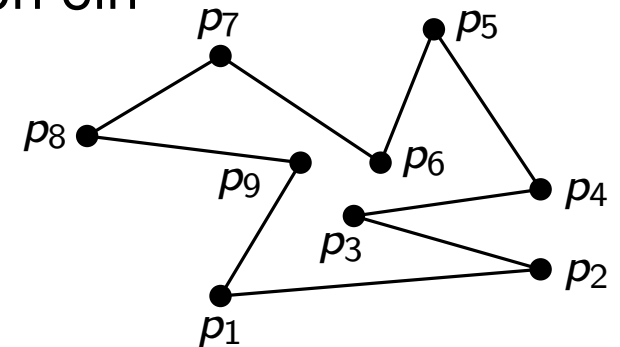
- füge Punkte entsprechend Reihenfolge im Polygon ein
- zu jedem Zeitpunkt: berechne konvexe Hülle der schon eingefügten Punkte



Wie geht es richtig?

Ansatz für einen korrekten Algorithmus

- füge Punkte entsprechend Reihenfolge im Polygon ein
- zu jedem Zeitpunkt: berechne konvexe Hülle der schon eingefügten Punkte



Ziel: baue einen $O(n)$ Algorithmus mit diesem Ansatz

Wie verändert sich die konvexe Hülle in einem Schritt?

Wie hilft da die Polygon-Reihenfolge?

Wie bekommen wir damit Laufzeit $O(n)$?