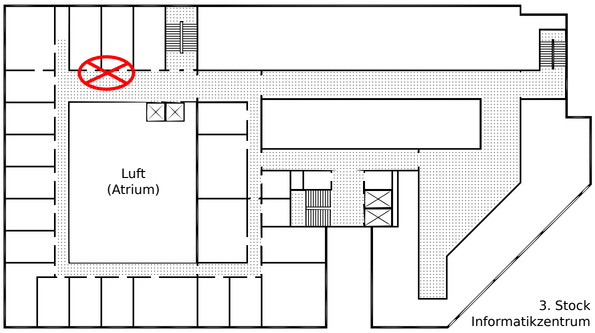
Abteilung Algorithmik **Winter 2019/20**
Institut für Betriebssysteme und Rechnerverbund
TU Braunschweig

Dr. Christian Scheffer
Christian Rieck

Algorithmische Geometrie
Übungsblatt 2 vom 18.11.2019

Die Abgabe der Lösungen zu Blatt 2 ist bis Montag, den 02.12.2019 um 11:30 Uhr im Hausaufgabenrückgabeschrank der Algorithmik möglich.

Bitte die Blätter vorne deutlich mit eigenem Namen sowie Matrikelnummer versehen und zusammenheften!



Hausaufgabe 1: Wende Jarvis' March auf die Punktmenge $\mathcal{P}$ aus Abbildung 1 an. Gib hierzu für jede Iteration der REPEAT-Schleife an, welcher neue Eckpunkt der konvexen Hülle gefunden wird. Gib außerdem innerhalb jeder Iteration der REPEAT-Schleife nach jeder Aktualisierung von p_{next} an, welcher Punkt p_i der Punkt p_{next} ist. Betrachte für diese Aufgabe folgende Veränderung des Algorithmus Jarvis' March:

- In jeder Iteration der REPEAT − UNTIL-Schleife wird p_{next} initial als der Punkt aus $\mathcal{P} \setminus \{p\}$ gewählt, der den kleinsten Index hat.
- In der FOR-Schleife werden die Punkte aus $\mathcal{P}$ in der Reihenfolge $\langle p_1, \ldots, p_n \rangle$ abgearbeitet.

(7 Punkte)

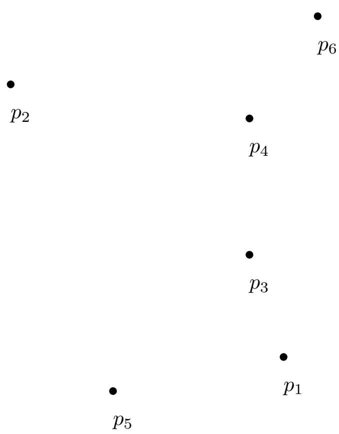


Abbildung 1: Die Punktmenge $\mathcal{P}$ für Aufgabe 1.

Hausaufgabe 2: Seien L_1 und L_2 zwei parallele Liniensegmente und sei o.B.d.A. L_1 länger als L_2. Gib eine Konstruktion an, um L_2 in sechs gleiche Teile zu teilen. Als Hilfsmittel ist nur ein Lineal ohne Messskala erlaubt. **(8 Punkte)**

Hausaufgabe 3: Die *konvexen Schichten* einer Punktmenge $\mathcal{P}$ sind geschachtelte konvexe Polygone mit den Punkten aus $\mathcal{P}$ als Knoten. Gib einen Algorithmus an, der die konvexen Schichten einer Punktmenge $\mathcal{P}$ mit n Punkten in $O(n^2)$ Zeit bestimmt. Begründe die Korrektheit und Laufzeit. **(5 Punkte)**

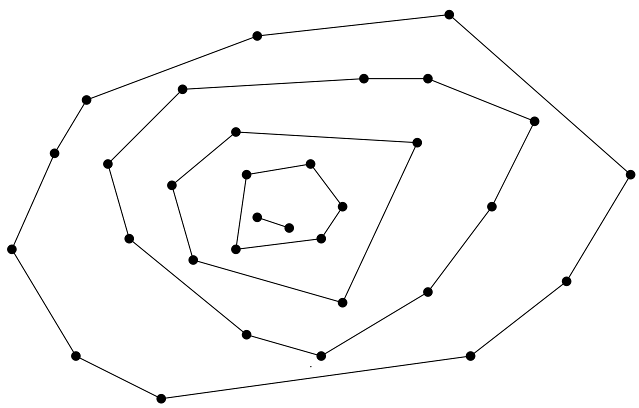


Abbildung 2: Eine Punktmenge $\mathcal{P}$ und ihre fünf konvexen Schichten.

Hausaufgabe 4: In dieser Aufgabe sollen die Details des in Kapitel 2 bearbeiteten Algorithmus QUICKHULL ausgearbeitet werden.

a) Gib an, wie man mit den in der Vorlesung besprochenen elementaren geometrischen Prädikaten das *Pivotelement* h bestimmen kann (verwende nicht explizit die Euklidische Distanzfunktion). Begründe, warum Dein Ansatz korrekt ist.

b) Analysiere die Laufzeit des resultierenden Algorithmus. Gib hierbei je eine Familie von Punktemengen (d.h. eine mit n parametrisierbare Konstruktionsvorschrift zum Erzeugen einer Konfiguration mit n Punkten) an, für die QUICKHULL $\Theta(n \log n)$ Zeit bzw. $\Theta(n^2)$ Zeit benötigt. Begründe dabei jeweils, warum die geforderte Laufzeit auch erreicht wird.

(5+5 Punkte)

Präsenzaufgabe 1: Wir möchten folgendes Problem lösen: Gegeben ein Polygon $\mathcal{P}$ mit n Eckpunkten, möglicherweise mit Löchern, und ein Punkt v im Inneren von $\mathcal{P}$. Berechne das Sichtbarkeitspolygon $\mathrm{Vis}(v)$, d.h., alle Punkte im inneren von $\mathcal{P}$ die von v aus sichtbar sind. $\mathrm{Vis}(v)$ bildet stets ein einfaches sternförmiges Polygon, es reicht daher die Eckpunkte dieses Polygons auszugeben. Ein Beispiel ist in Abbildung 3 gegeben.

Dieses Problem lässt sich mit einem Sweep-Line-Verfahren lösen. Beschreibe das Verfahren und beantworte dabei folgende Fragen:

- Wie sieht die Sweep-Line aus? Tipp: Sie bewegt sich **nicht** von links nach rechts.
- Was sind die Events und in welcher Reihenfolge werden sie bearbeitet?
- Was wird im Status der Sweep-Line gespeichert?
- Welche Änderung im Status hat eine Ausgabe eines neuen Eckpunktes von $\mathrm{Vis}(v)$ zur Folge.
- Was ist die Laufzeit?

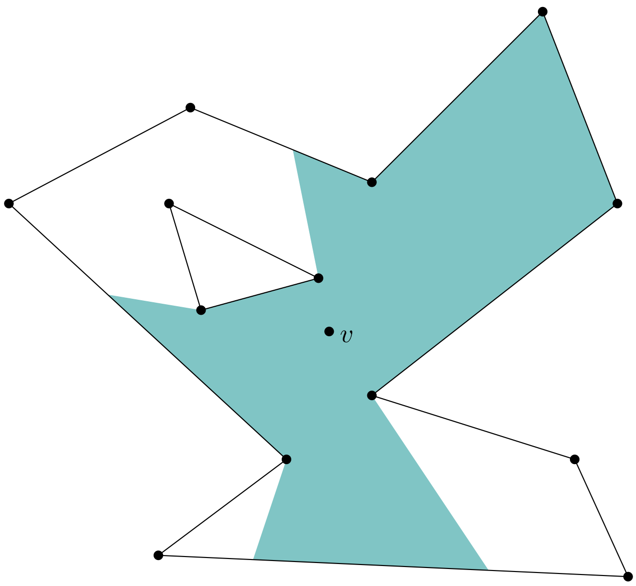


Abbildung 3: Ein Polygon mit einem Loch. Das Sichtbarkeitspolygon von v ist farblich hervorgehoben.