

## Übungsblatt 3

Julius Auer, Alexa Schlegel

---

### Aufgabe 1 (Graham-Scan):

Implementieren Sie den Graham-Scan. Benutzen Sie dabei aus Gründen besserer Laufzeit und zur Vermeidung von Rundungsfehlern möglichst nur die Grundrechenarten  $+$ ,  $-$ ,  $\times$ ,  $\div$ . Eine Umrechnung in Polarkoordinaten, wie in der Vorlesung beschrieben, ist nicht nötig, um die Strahlen nach Steigung zu sortieren.

Im Folgenden wird der Algorithmus beschrieben, welcher implementiert wurde. Er orientiert sich an der in *Introduction to Algorithms*<sup>1</sup>. Der Stack  $S$  wird verwendet, um mögliche Kandidaten der konvexen Hülle abzulegen. Terminiert der Algorithmus so enthält  $S$  die Knoten der konvexen Hülle, gegen den Urzeigersinn.

---

#### Algorithm 1 Graham-Scan( $Q$ )

---

```
1: sei  $p_0 \in Q$  mit minimaler  $y$ -Koordinate: bei gleicher  $y$ -Koordinate minimale  $x$ -Koordinate
   verwenden
2: sei  $p_1, p_2, \dots, p_n$  die restlichen Punkte aus  $Q$ :
   sortiert nach Polarkoordinaten
   gegen den Urzeigersinn
   bei gleichem Winkel, nur den Punkt behalten der den größten Abstand von  $p_0$  hat
   (TODO schauen wie das ohne Polarkoordinaten auch geht, evtl. Kreuzprodukt oder so)
3: PUSH( $p_0, S$ )
4: PUSH( $p_1, S$ )
5: PUSH( $p_2, S$ )
6: for  $i = 3$  to  $n$  do
7:   while Winkel (NEXT-TO-TOP( $S$ ), TOP( $S$ ),  $p_i$ ) ist keine Linkskurve do
8:     POP( $S$ )
9:   end while
10:  PUSH( $p_i, S$ )
11: end for
12: return  $S$ 
```

---

### Aufgabe 2 (inkrementelle Konstruktion der konvexen Hülle):

TODO

### Aufgabe 3 (untere Schranke):

TODO

---

<sup>1</sup>Introduction to Algorithms, Second Edition, S. 949