

## Übungsblatt 1

### Aufgabe 1: Closest Pair per Divide and Conquer (4 Punkte)

Beschreiben Sie, wie der Abstand eines dichtesten Punktepaars von  $n$  Punkten in der Ebene mit dem *Divide and Conquer*-Verfahren in Zeit  $O(n \log n)$  bestimmt werden kann

### Aufgabe 2: Maximale Produktteifolge (4 Punkte)

Zu der Folge  $a_1, a_2, \dots, a_n$  positiver reeller Zahlen soll eine zusammenhängende Teilfolge  $a_j, \dots, a_k$  gefunden werden, so dass das Produkt  $\prod_{i=j}^k a_i$  maximal wird.

- Modifizieren Sie den Algorithmus zur Bestimmung der maximalen Teilsumme derart, dass er das Problem des maximalen Teilproduktes löst.
- Beschreiben Sie, wie durch eine geeignete Transformation der Eingabedaten das Problem des maximalen Teilproduktes auf das Problem der maximalen Teilsumme zurückgeführt werden kann. Geben Sie hierfür geeignete Abbildungen  $f$  und  $g$  an, so dass sich das maximale Teilprodukt der Folge  $a_1, a_2, \dots, a_n$  mit dem folgenden Verfahren bestimmen lässt:
  - $M$  wird als maximale Teilsumme der Folge  $f(a_1), \dots, f(a_n)$  berechnet.
  - Es erfolgt die Ausgabe von  $g(M)$ .

### Aufgabe 3: Dominierende Liniensegmente (4 Punkte)

Gegeben seien  $n$  horizontale und disjunkte Liniensegmente, wobei die  $x$ -Werte aller Endpunkte paarweise verschieden sind. Zu jedem Liniensegment  $s$  werden diejenigen Liniensegmente gesucht, welche direkt unterhalb von  $s$  liegen, d. h. eine vertikale Gerade schneidet die beiden Liniensegmente aber kein anderes dazwischen.

Formulieren Sie einen  $O(n \log n)$  *Sweep-Algorithmus*, der zu jedem Liniensegment  $s$  alle anderen berichtet, die von  $s$  in dem beschriebenen Sinne dominiert werden, begründen Sie, warum Ihr Algorithmus die gegebene Laufzeit hat und zeigen Sie, dass die Laufzeit optimal ist.

### Aufgabe 4: Untere Schranken (4 Punkte)

Gegeben seien  $n$  reelle Zahlen  $x_1, \dots, x_n$  und ein  $\varepsilon > 0$ . Gefragt ist, ob es Indizes  $i \neq j$  gibt, so dass  $|x_i - x_j| < |i - j|\varepsilon$  gilt. Geben Sie eine möglichst gute, untere Laufzeitschranke für dieses Problem in Abhängigkeit von  $n$  an.