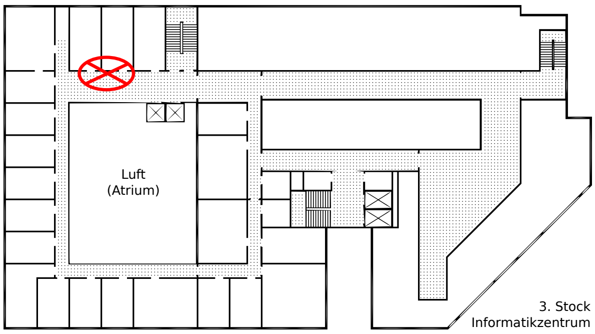
Abteilung Algorithmik **Winter 2017/2018**
Institut für Betriebssysteme und Rechnerverbund
TU Braunschweig

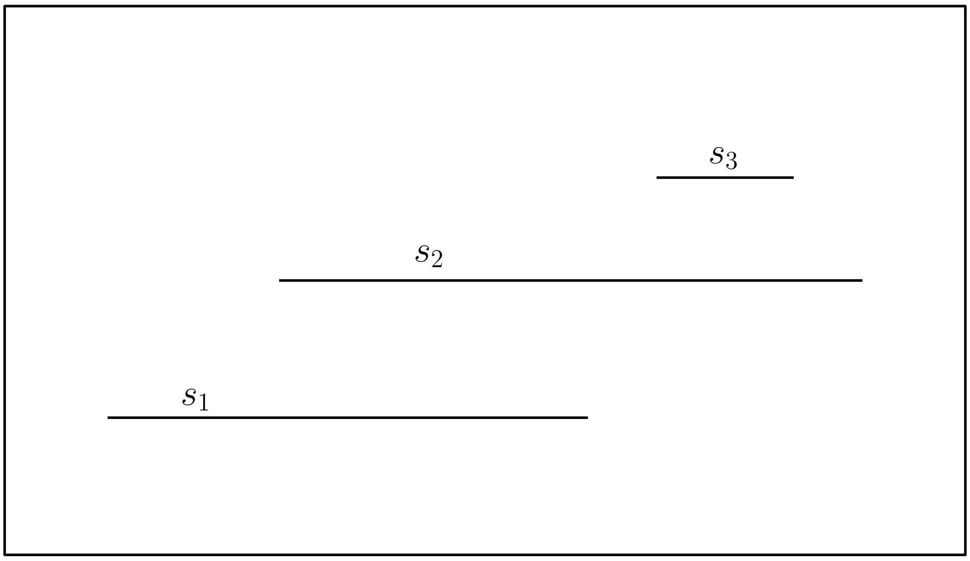
Dr. Christian Scheffer
Andreas Haas

Algorithmische Geometrie
Übung 4 vom 15. 12. 2017

Abgabe der Lösungen bis zum Freitag, den 12. 01. 2018 um 11:30 im Hausaufgabenrückgabeschrank.



Bitte die Blätter vorne deutlich mit eigenem Namen sowie Matrikel- und Gruppennummer versehen!



Aufgabe 1 (Planar Point Location - Trapezoidal Map): Füge die Segmente (s_1, s_2, s_3) entsprechend ihrer Reihenfolge ein.

a) Gib $\mathcal{T}(S_i)$ und $D(S_i)$ für $i \in \{1, 2, 3\}$ separat an.

Beachte, dass nicht alle Pfade in $D(S_3)$ valide Suchpfade sind. Sei $\mathcal{D}$ die maximale Länge aller Pfade in $D(S)$ und $\mathcal{L}$ die maximale Länger aller validen Suchpfade.

b) Markiere einen Pfad der Länge $\mathcal{D}$ in $D(S_3)$.

c) Markiere einen validen Suchpfad der Länge $\mathcal{L}$ in $D(S_3)$.

Bonus (SCHWIERIG):

- Sei n eine Quadratzahl, d.h., $n = k^2$ für ein $k \in \mathbb{N}$. Skizziere eine Menge von n Segmenten S und gib eine Einfügereihenfolge an, sodass die Länge des längstes Pfades in $D(S)$ in $\Omega(n)$ liegt, aber die Länge des längsten gültigen Suchpfades in $O(\sqrt{n})$.
- Sei n eine Zweierpotenz, d.h., $n = 2^k$ für ein $k \in \mathbb{N}$. Skizziere eine Menge von n Segmenten S und gib eine Einfügereihenfolge an, sodass die Länge des längstes Pfades in $D(S)$ in $\Omega(n)$ liegt, aber die Länge des längsten gültigen Suchpfades in $O(\log n)$.

(5 + 5 Bonus Punkte)

Aufgabe 2 (Trapezoidal Map - Worst-Case Complexity):

a) Gib eine Menge von n Segmenten S und eine Einfügereihenfolge an, sodass die Größe der Suchstruktur $D(S)$ in $\Omega(n^2)$ liegt.

b) Gib eine Menge von n Segmenten S und eine Einfügereihenfolge an, sodass die Länge des längsten Suchpfades in $D(S)$ in $\Omega(n)$ liegt.

(5 Punkte)

Aufgabe 3 (Randomized Incremental Construction - Backward Analysis): Betrachte eine randomisierte inkrementelle Konstruktion eines binären Suchbaumes T. Zu Beginn wird eine zufällige Reihenfolge für das Einfügen der Elemente bestimmt, diese werden anschließend und ohne weitere Balancierung-Operationen eingefügt. Sei eine Menge S aus n Elementen und ein weiterer Punkt $p \notin S$ gegeben. Zeige, dass die erwartete Zeit um p in T zu finden[1] in $O(\log n)$ liegt.

Hinweis: Betrachte den Pfad zu p in T während der Konstruktion. Wie hoch ist die Wahrscheinlichkeit, dass sich die Länge durch das Einfügen eines Elementes erhöht?

(5 Punkte)

Aufgabe 4: Entwirf einen Algorithmus, der in $\mathcal{O}(n \log n)$ Zeit zu einer endlichen Menge $\mathcal{S} \subset \mathbb{R}^2$ von Punkten ein Punktpaar mit maximalem Abstand bestimmt.

Beweise, warum Dein Algorithmus korrekt ist und die geforderte Laufzeit hat.

Hinweis: Wenn Du eine Idee für den Lösungsansatz benötigst, informiere Dich über das Stichwort *rotating calipers* (z.B. unter der URL `http://cgm.cs.mcgill.ca/~godfried/research/calipers.html`).

(5 Punkte)

[1] D.h. die Position in T an der sich p befinden müsste.