Prof. Dr. Ir. Joost-Pieter Katoen

Jonathan Heinen, Sabrina von Styp

Hinweise:

- Die Übungsblätter sind in Gruppen von je 3 Studierenden der gleichen Kleingruppenübung zu bearbeitet.
- Die Lösungen müssen bis Montag, den 26. Juli um 11:00 Uhr in die Übungskasten eingeworfen werden. Sie finden die Kästen am Eingang Halifaxstr. des Informatikzentrums (Ahornstr. 55).
- Namen und Matrikelnummern, sowie die Nummer der Übungsgruppe sind auf jedes Blatt zu schreiben.

Aufgabe 1 (Longest Common Subsequence): (5 Punkte)

Bestimmen Sie, entsprechend dem in der Vorlesung vorgestellten Verfahren, die Longest Common Subsequence der Wörter `WELTMEISTER` und `ELFMETER`. Geben Sie hierzu die berechnete Matrix an und kennzeichnen Sie den Pfad, der zur Rekonstruktion des Ergebnisses genutzt wird.

Aufgabe 2 (Levenshtein-Distanz): (5 + 4 + 5 + 3 Punkte)

Die Levenshtein-Distanz zwischen zwei Zeichenketten ist die minimale Anzahl von Einfüge-, Lösch- und Ersetz-Operationen die nötig sind, um die eine Zeichenkette u in die andere v umzuwandeln.
Als Beispiel betrachten wir die Zeichenketten $u =$ `Tier` und $v =$ `Tor`. In einem ersten Schritt können wir u durch das Ersetzen von `e` durch `o` in `Tior` umwandeln. In einem zweiten Schritt erhalten wir dann durch Löschen des Zeichen `i` das Wort $v =$ `Tor`. Da es keine kürzeren Umwandlungssequenzen gibt beträgt die Levenshtein-Distanz zwischen `Tier` und `Tor` 2.
Wie lässt sich die Levenshtein-Distanz zwischen zwei beliebigen Zeichenketten berechnen?

a) Geben Sie eine Rekursionsgleichung zur Berechnung der Levenshtein-Distanz zwischen den Zeichenketten u und v an.

b) Nutzen Sie die Rekursionsgleichung aus **a)** bottom-up, um eine $|u| \times |v|$ Matrix für die Überführung von `RHABARBER` in `BARBAREI` zu füllen und lesen Sie aus dieser die Levenshtein-Distanz ab.

c) Nutzen Sie die Rekursionsgleichung aus **a)** für die Implementierung der Methode `levenshtein(char u[n], char v[m], int n, int m)`, die – entsprechend dem Verfahren aus **b)** – die Levenshtein-Distanz zwischen den Zeichenketten u und v bestimmt.

d) Bestimmen Sie die Laufzeit der von Ihnen erstellten Methode in Abhängigkeit von n und m, den Längen der beiden Zeichenketten.

Aufgabe 3 (Graham-Algorithmus): (2 + 4 Punkte)

Gegeben seien die folgenden Punkte:

$$(12,4),\ (7,7),\ (3,13),\ (6,13),\ (4,2),\ (4,14),\ (9,10),\ (4,13),\ (8,12),\ (2,12)$$

a) Bestimmen Sie den Punkt p_1 mit der minimalsten x-Koordinate und sortieren sie die restlichen Punkte nach aufsteigendem Polarwinkel.

b) Bestimmen Sie, mit Hilfe des Graham-Algorithmus die konvexe Hülle der oben angegebenen Punkte. Geben Sie alle berechneten Determinanten an, sowie den jeweils aktuellen Zustand des Stacks.

Aufgabe 4 (Ghostbusters): (5 + 5 Punkte)

Ein Gruppe von n Geisterjägern kämpft gegen n Geister. Jeder Geisterjäger ist mit einem Protonenwerfer bewaffnet. Der Strahl eines Protonenwerfers verläuft auf einer geraden Linie und bricht ab sobald er einen Geist trifft. Die Geisterjäger entscheiden, die Geister unter sich aufzuteilen, um dann gleichzeitig auf den jeweils zugeteilten Geist zu schiessen. Bekannterweise ist es sehr gefährlich, wenn sich Protonenstrahlen kreuzen, weshalb die Geisterjäger die Geister so unter sich aufteilen müssen, dass sich die Strahlen nicht kreuzen.
Gehen Sie davon aus, dass die Position jedes Geisterjägers, sowie jedes Geistes ein fester Punkt einer Ebene ist und keine drei Positionen auf einer gemeinsamen Gerade liegen.

a) Zeigen Sie, dass es eine Gerade gibt, die durch einen Geisterjäger und einen Geist verläuft, sodass auf jeder Seite die Anzahl der Geisterjäger gleich der Anzahl von Geistern ist.

Beschreiben Sie, wie eine solche Gerade in Zeit $O(n \log n)$ gefunden werden kann.

b) Geben Sie einen Algorithmus mit Laufzeit $O(n^2 \log n)$ an, der die Geister so auf die Geisterjäger aufteilt, dass sich die Protonenstrahlen nicht schneiden.