Algorithmen und Programmierung 3, WS 2003/2004 — 12. Übungsblatt

Freiwillige Abgabe bis Donnerstag, 12. Februar 2004

70. (7 Punkte) Ein Iterator für Mengen.
    (a) (4 Punkte) Implementieren Sie den *Iterator*, den Sie in Aufgabe 52 spezifiziert haben. Sie können von der in der Vorlesung besprochenen Implementierung für Mengen mit bis zu 100 Elementen[1] ausgehen.
    (b) (3 Punkte) Geben Sie die Abstraktionsfunktion an.
    (c) (5 Zusatzpunkte) Beweisen Sie die Korrektheit Ihrer Implementierung.

71. (3 Punkte) Gegeben sei ein vollständiger[2] Graph $G = (V, E)$, dessen Knoten V Punkte in der Ebene sind, und wo die Kantenlängen den (Euklidischen) Abständen zwischen den Punkten entsprechen. Beweisen Sie:
    (a) (3 Punkte) In einem kürzesten spannenden Baum können sich nie zwei Kanten kreuzen.
    (b) (5 Zusatzpunkte) In einem kürzesten-Wege-Baum (mit einem beliebigen Startknoten) können sich nie zwei Kanten kreuzen.

    Bleiben diese Aussagen auch gültig, wenn der Graph nicht vollständig ist?

72. (0 Punkte) Für zwei Wörter x und y der Länge n ist x eine *zyklische* Verschiebung von y, wenn man $x = ab$ und $y = ba$ für zwei Wörter a und b schreiben kann. Wie kann man in linearer Zeit feststellen, ob x eine zyklische Verschiebung von y ist? (Man kann diese Frage auf das Teilwortproblem zurückführen.)

73. (5 Punkte) Wie kann man den Algorithmus von Knuth, Morris und Pratt so erweitern, dass er *alle* Vorkommen eines Musters x in einem Text w findet? (Man benötigt eventuell den Wert h_{m+1} der Verschiebefunktion, für $m = |x|$.)

74. (0 Punkte) Die verbesserte Verschiebefunktion zum Suchen von Zeichenketten ist folgendermaßen definiert:

    $$h[i] = \max \{\, k \mid 1 \le k < i,\ p_1 \dots p_{k-1} = p_{i-k+1} \dots p_{i-1} \text{ und } p_k \ne p_i \,\} \cup \{0\}$$

    (a) Berechnen Sie die verbesserte Verschiebefunktion der Muster aus Aufgabe 68.
    (b) Zeigen Sie, dass beim Suchen mit der verbesserten Verschiebefunktion auf keinen Fall mehr Vergleiche der Form $p_i = s_j$ durchgeführt werden als mit der ursprünglichen Verschiebefunktion (ohne die Bedingung „$p_k \ne p_i$“). Gilt dies auch, wenn man den Aufwand an Vergleichen beim Berechnen der Verschiebefunktion mit berücksichtigt? Finden Sie ein Beispiel, bei dem tatsächlich weniger Vergleiche notwendig sind.
    (c) Schreiben Sie einen Algorithmus zum Berechnen der verbesserten Verschiebefunktion.

75. (0 Punkte) Wie kann man aus der Verschiebefunktion des Wortes $xw\$$ berechnen, ob x ein Teilwort von y ist? (Hier ist \$ irgendein Buchstabe.)

76. (0 Punkte) Welche Bedeutung hat die Flächenformel $\frac{1}{2} \cdot |\sum_i (x_i y_{i+1} - y_i x_{i+1})|$, wenn die Folge der Punkte $(x_i; y_i)$ gar kein Polygon beschreibt, weil sich zum Beispiel Kanten kreuzen?

77. (a) (5 Punkte) Berechnen Sie die Fläche des Fünfecks[3] mit den Ecken $(-1{,}3; 10000{,}12)$, $(-0{,}253; 10000{,}47)$, $(0{,}69; 10000{,}33)$, $(1{,}529; 10002{,}12)$, $(-0{,}783; 10001{,}05)$ mit der Formel aus der vorigen Aufgabe. Berechnen Sie auch den Flächeninhalt des um den Vektor $(0; -10000)$ verschobenen Fünfecks. Welches Ergebnis halten Sie für das genauere? Begründen Sie Ihre Antwort.
    (b) (0 Punkte) Was passiert, wenn man das Fünfeck um den Vektor $(10000; -10000)$ verschiebt? Wie erklären Sie diese Ergebnisse?

[1] `http://www.inf.fu-berlin.de/~rote/Lere/2003-04-WS/Algorithmen+Programmierung3/Menge.java`

[2] Ein Graph ist *vollständig*, wenn zwischen allen Paaren von Knoten eine Kante verläuft.

[3] `http://www.inf.fu-berlin.de/~rote/Lere/2003-04-WS/Algorithmen+Programmierung3/5eck`