Johannes Blömer
u.v.a.m.

Paderborn, den 22. Juli
Abgabe ??. August 2004, 11:00 Uhr, Kästen Flur D3

Übungen zur Vorlesung

Datenstrukturen und Algorithmen

SS 2004

Blatt 14

AUFGABE 1 (6 Punkte)**:**
Wir schreiben $C(n,k)$ für den Binomialkoeffizienten $\binom{n}{k}$. Die Binomialkoeffizienten sind folgendermaßen rekursiv definiert:

$$C(n,k) = 1 \qquad \text{für } k = 0 \text{ und } k = n$$

$$C(n,k) = C(n-1,k) + C(n-1,k-1) \qquad \text{für } 0 < k < n.$$

Benutzen Sie diese rekursive Definition und die Technik der dynamischen Programmierung, um einen Algorithmus zu entwerfen, der bei Eingabe n, k den Binomialkoeffizienten $C(n,k)$ berechnet. Analysieren Sie die Laufzeit Ihres Algorithmus.

AUFGABE 2 (8 Punkte)**:**
Bei einem gerichteten, gewichteten Graphen (G,w) mit $G = (V,E)$, ist w eine Funktion, die jeder Kante $(u,v) \in E$ einen positiven reellen Wert zuordnet. Wir bezeichnen den Wert $w((u,v))$ auch mit w_{uv}. Wir betrachten nun das Problem *All-Pairs-Shortest-Path (APSP)*. Hierbei ist ein gerichteter, gewichteter Graph (G,w) mit $G = (V,E)$ gegeben. Gesucht ist für alle Paare $(u,v) \in V^2$ die Länge eines kürzesten Pfades in G, der von u nach v führt. Die Länge eines Pfades ist dabei die Summen der Gewichte der Kanten auf dem Pfad. Existiert kein Pfad von u nach v, so ist die Länge eines kürzesten Pfades von u nach v definiert als ∞.
Benutzen Sie dynamische Programmierung, um einen Algorithmus für APSP zu entwerfen.
Hinweis: Nehmen Sie an, dass die Knotenmenge V des Graphen G gegeben ist durch $\{1,\ldots,n\}$. Für je zwei Knoten u,v und $k \in \{1,\ldots,n\}$ definieren Sie dann $d_{uv}^{(k)}$ als die Länge eines kürzesten Pfades von u nach v auf dem abgesehen von u und v nur Knoten aus $\{1,\ldots,k\}$ liegen. Drücken Sie $d_{uv}^{(k)}$ rekursiv mit Hilfe von Werten der Form $d_{xy}^{(k-1)}$ aus. Benutzen Sie dann diese Rekursion, um Ihren Algorithmus zu entwerfen.