Diskrete Mathematik für Informatikstudien
Sommersemester 2022

12. Übungsblatt (14.6.2022)

Beispiel 12.1. Zeichnen Sie den Baum T mit $V(T) = \{1, \ldots, 9\}$ und

$$E(T) = \big\{\{1,5\}, \{1,8\}, \{1,9\}, \{2,3\}, \{2,7\}, \{2,8\}, \{4,7\}, \{6,8\}\big\}$$

und ermitteln Sie seinen Prüfer-Code. Überprüfen Sie anschließend, dass die Behauptung

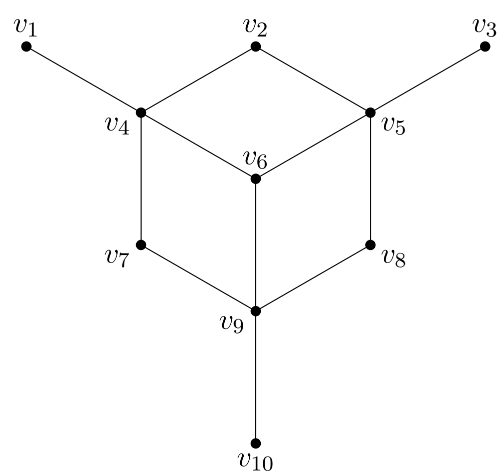
Jeder Knoten x von T kommt genau $(d(x) - 1)$-mal im Prüfer-Code von T vor.

für diesen Baum korrekt ist.

Beispiel 12.2. Der Graph G mit Knotenmenge $\{v_1, \ldots, v_{10}\}$ und Kantenmenge

$$\big\{\{v_1, v_4\}, \{v_2, v_4\}, \{v_2, v_5\}, \{v_3, v_5\}, \{v_4, v_6\}, \{v_4, v_7\},\\ \{v_5, v_6\}, \{v_5, v_8\}, \{v_6, v_9\}, \{v_7, v_9\}, \{v_8, v_9\}, \{v_9, v_{10}\}\big\}$$

kann wie folgt gezeichnet werden.



Konstruieren Sie einen BFS-Spannbaum und einen DFS-Spannbaum von G, jeweils ausgehend von der Wurzel v_1. Verwenden Sie dabei folgende Regel: Wann immer im Algorithmus mehrere Nachbarn des aktuell betrachteten Knoten zur Auswahl stehen, wählen Sie den Knoten mit dem kleinsten Index.

Notieren Sie für jeden Schritt der Algorithmen die Liste zu Beginn dieses Schrittes, die zum Baum hinzugefügte Kante (falls in dem Schritt eine Kante hinzugefügt wird) und die Änderung, die an der Liste vorgenommen wird (z.B. „füge v_3 am Ende/am Anfang hinzu“ oder „entferne v_7“).

Beispiel 12.3. Gegeben ist ein Graph $G = (V, E)$ und ein Knoten $x \in V$. Wie in Beispielen 11.3 und 11.4 bezeichnen wir für $y \in V(G)$ den Abstand von x und y mit $d(x, y)$ und schreiben $D_i := \{y \in V \mid d(x, y) = i\}$. Wir führen nun BFS beginnend bei der Wurzel x durch. Zeigen Sie, dass durch den Algorithmus zuerst alle Knoten in D_1 zum BFS-Baum hinzugefügt werden, dann alle Knoten in D_2, dann alle Knoten in D_3 und so weiter. Folgern Sie hieraus, dass jeder Knoten $y \in D_i$ auch in T den Abstand i von x hat.

Beispiel 12.4. Wir betrachten den Graphen G mit Knotenmenge $\{v_1, \dots, v_8\}$, Kantenmenge

$$\big\{\{v_1, v_2\}, \{v_1, v_5\}, \{v_1, v_8\}, \{v_2, v_3\}, \{v_2, v_8\}, \{v_3, v_4\},\\ \{v_3, v_7\}, \{v_4, v_5\}, \{v_4, v_6\}, \{v_5, v_6\}, \{v_6, v_7\}, \{v_7, v_8\}\big\}$$

und Kantengewichten $w(\{v_i, v_j\}) = i+j$. Zeichnen Sie den Graphen (inklusive Kantengewichten) und führen Sie den Algorithmus von Kruskal durch, um einen Spannbaum von minimalem Gesamtgewicht zu finden. Ist dieser minimale Spannbaum eindeutig?

Begründen Sie für jeden Schritt des Algorithmus, warum die gerade betrachtete Kante hinzugefügt oder nicht hinzugefügt wird.

Beispiel 12.5. Wir betrachten die folgenden Varianten des Algorithmus von Kruskal. In beiden Fällen werden wie im Algorithmus von Kruskal zuerst alle Kanten als $e_1, \dots, e_m$ sortiert, sodass $w(e_1) \leq \ldots \leq w(e_m)$. (Zur Vereinfachung der Notation nehmen wir an, dass G genau n Knoten und m Kanten hat.)

(A) Wir setzen $F = E(G)$. Für $i = m$ hinunter bis $i = 1$ führen wir nun folgende Operation durch: Falls $(V(G), F \setminus \{e_i\})$ zusammenhängend ist, entferne e_i aus F. Die am Ende verbleibenden Kanten in F bilden die Kanten eines Spannbaumes von G.

(B) Wir beginnen an einem Knoten x_0 und bezeichnen mit T_0 den Baum, der nur aus x_0 besteht. Für $i = 1, \dots, n-1$ führen wir folgende Operation durch: Wähle die Kante e_{j_i} mit dem kleinsten Index j_i, die einen Endknoten in T_{i-1} und einen Endknoten x_i außerhalb von T_{i-1} hat. Füge x_i und e_{j_i} zu T_{i-1} hinzu und nenne diesen neuen Baum T_i. Der letzte so konstruierte Baum T_{n-1} ist ein Spannbaum von G.

Führen Sie beide Algorithmen für den Graphen aus Beispiel 12.4 durch, wobei Sie bei Algorithmus B bei $x_0 = v_5$ beginnen. Begründen Sie wieder in jedem Schritt des Algorithmus, warum die betreffende Kante entfernt bzw. hinzugefügt wird.