Lehrstuhl für Informatik I
Algorithmen und Komplexität
Universität Würzburg

Würzburg, den 23. Januar 2025

Prof. Dr. Alexander Wolff

4. Präsenzübungsblatt zur Vorlesung Algorithmen und Datenstrukturen (Winter 2024/25)

Aufgabe 1 – Faire Geschenke

Sie sind Mutter/Vater von Zwillingen, deren Geburtstag naht. Sie haben eine Menge an n Geschenken gekauft und möchten diese nun fair verteilen. Dabei hat jedes Geschenk i einen bestimmten ganzzahligen positiven Wert w_i. Alle Geschenke zusammen addieren sich zum dem Wert W. Wir nehmen an, dass dieser gerade ist. Die Aufteilung der Geschenke ist genau dann fair, wenn jeder Zwilling insgesamt den gleichen Wert an Geschenken erhält.

Geben Sie einen Algorithmus an, der in Zeit $O(n \cdot W)$ feststellt, ob eine faire Aufteilung aller Geschenke möglich ist.

Aufgabe 2 – Pfade in kreisfreien Graphen

Gegeben sei ein gerichteter kreisfreier Graph $G = (V, E)$, zwei Knoten $s, t \in V$ und eine Menge $W \subset V$ von k Knoten, wobei $s, t \notin W$. Wir bezeichnen einen s-t-Pfad P als *zulässig*, wenn er durch alle Knoten in W geht. Dabei ist es egal, in welcher Reihenfolge die Knoten in W durchlaufen werden.

a) Seien P_1 und P_2 zwei zulässige s-t-Pfade. Zeigen Sie, dass P_1 und P_2 die Knoten in W in derselben Reihenfolge durchlaufen.

b) Gesucht ist ein Algorithmus, der ein Feld $A[1..(k+2)]$ liefert, das den Knoten s, die Knoten in W und den Knoten t enthält. Falls es einen zulässigen s-t-Pfad P gibt, so sollen die Knoten in A entsprechend des Verlaufs von P geordnet sein. Falls es keinen zulässigen s-t-Pfad gibt, so können die Knoten in beliebiger Reihenfolge ins Feld A geschrieben werden.

   Geben Sie den Algorithmus

   getVertexOrder(DirectedAcyclicGraph G, Vertex s, Vertex t, Array of Vertices W)

   in Pseudocode an, der das Feld A in $\Theta(|V| + |E|)$ Zeit berechnet und zurückgibt. Beschreiben Sie Ihren Algorithmus auch in Worten, und begründen Sie, warum er korrekt ist.

Hinweis: Um für einen gegebenen Knoten $v \in V$ in konstanter Zeit testen zu können, ob $v \in W$, können Sie davon ausgehen, dass es ein Knotenattribut $\mathtt{flag}$ *vom Typ boolean gibt, wobei* $v.\mathtt{flag} == \mathtt{true} \Leftrightarrow v \in W$.

c) Ergänzen Sie den folgenden Algorithmus, so dass er einen zulässigen s-t-Pfad als Liste von Knoten zurückgibt, falls es solch einen Pfad gibt. Falls es keinen zulässigen s-t-Pfad gibt, soll der Algorithmus $\mathtt{nil}$ zurückgeben. Begründen Sie, warum der Algorithmus korrekt ist.

getPath(DirectedAcyclicGraph G, Vertex s, Vertex t, Array of Vertices W)
P = new List()
A = getVertexOrder(G, s, t, W) // Liefert ein Feld mit s, mit den Knoten in W in der richtigen Reihenfolge, wenn es eine solche gibt, und mit t.
for $i = A.\mathrm{length}$ ***downto*** 2 **do**
    // Ergänze Pfad P um Knoten zwischen $A[i-1]$ und $A[i]$
    // (ohne $A[i-1]$ aber einschließlich $A[i]$).
    // Gib $\mathtt{nil}$ zurück, wenn es von $A[i-1]$ nach $A[i]$ keinen Pfad gibt.
P.Insert($A[1]$)
return P

d) Hat Ihr Algorithmus aus Aufgabenteil c) die Laufzeit $O(|V| + |E|)$? Falls nicht, beschreiben Sie in Worten, wie man diese Laufzeit durch Verbesserung Ihres Algorithmus erreichen kann.

e) Gegeben sei ein gerichteter Graph $G = (V, E)$, der nicht notwendigerweise kreisfrei ist, zwei Knoten $s, t \in V$ und eine Folge von Knoten $W = (w_1, w_2, \ldots, w_n)$. Zeigen Sie, dass im schlechtesten Fall jeder s-t-Pfad, der die Knoten in W durchläuft und dabei die gegebene Reihenfolge einhält, Länge $\Omega(|V|^2)$ hat. Beachten Sie, dass ein Pfad einen Knoten mehrfach durchlaufen kann.

Aufgabe 3 – Leichte Kanten und Schnitte

Sei $G = (V, E)$ ein Graph mit Kantengewichtsfunktion $w : E \to \mathbb{R}^+$. Sei weiter $T = (V, E_T)$ ein minimaler Spannbaum von G bezüglich w. Zeigen Sie: für jede Kante $\{u, v\} \in E_T$ gibt es einen Schnitt, für den $\{u, v\}$ *leicht* ist.

Diese Aufgaben werden eventuell gemeinsam in den Übungen am 4. und 5. Februar 2025 gelöst. Sie brauchen Sie nicht vorher zu lösen und auch nicht abzugeben.