

9. Übungsblatt zur Vorlesung Algorithmen und Datenstrukturen (Winter 2019/20)

Aufgabe 1 – Wandern im Gebirge

Sie planen eine gefährliche Wanderung in einem gebirgigen Terrain, bei der Sie immer wieder über tiefe Schluchten springen müssen. Ihnen steht eine Wanderkarte zur Verfügung, auf der Berggipfel und Wegabschnitte eingezeichnet sind. Ein Wegabschnitt verbindet immer zwei Berggipfel und enthält genau eine Schlucht, die man auf dem Wegabschnitt überwinden muss. Zu jedem Wegabschnitt ist zusätzlich die Breite der zu überwindenden Schlucht in Zentimetern angegeben. Einige Wegabschnitte kann man nur in eine Richtung gehen, weil man z.B. herunter springen muss.

Sie starten Ihre Wanderung auf einem Berggipfel s und möchten herausfinden, wie weit man in der Lage sein muss zu springen, um jeden Berggipfel von s aus zu erreichen. (Ein Berggipfel t ist für Sie von s aus erreichbar, wenn es einen Weg von s nach t gibt, bei der Sie jede Schlucht überspringen können.)

- a) Noch aus dem Sportunterricht in der Schule wissen Sie, dass Sie lediglich ℓ Zentimeter weit springen können, Sie also über Schluchten der Breite größer als ℓ nicht springen können.

Sie möchten herausfinden, welche Berggipfel Sie von s aus erreichen können. Modellieren Sie zunächst das Problem als ein Problem auf einem Graphen. Geben Sie anschließend einen Algorithmus in Worten an, der dieses Problem löst und eine Worst-Case-Laufzeit hat, die linear in der Anzahl der Berggipfel und Wegabschnitte ist.

6 Punkte

- b) Sie möchten vor Ihrer Wanderung trainieren und Ihre Sprungweite verbessern. Hierzu möchten Sie für jeden Berggipfel t herausfinden, wie weit Sie in der Lage sein müssen zu springen um t von s zu erreichen.

Geben Sie in Worten einen Algorithmus an, der dieses Problem möglichst effizient löst. Geben Sie seine Worst-Case-Laufzeit in Abhängigkeit von der Anzahl der Berggipfel und der Anzahl der Wegabschnitte an.

4 Punkte

Aufgabe 2 – Tiefensuche

Geben Sie für jedes der geforderten Beispiele den Graphen und das Ergebnis einer Tiefensuche in Form der resultierenden Bäume sowie der Entdeckungs- und Abschlusszeiten an. In dieser Aufgabe sind keine Selbst- oder Mehrfachkanten erlaubt.

a) Geben Sie ein Beispiel an, das folgende Behauptung widerlegt:

Sei G ein gerichteter Graph, der einen Pfad von u nach v enthält, und sei $u.d < v.d$ das Resultat einer Tiefensuche in G . Dann folgt, dass v im Tiefensuchbaum ein Nachkomme von u ist (d.h. es gibt in diesem Baum einen u - v -Pfad). **2 Punkte**

b) Geben Sie ein Beispiel an, das folgende Behauptung widerlegt:

Sei G ein gerichteter Graph, der einen Pfad von u nach v enthält. Für die Entdeckungs- und Abschlusszeiten jeder Tiefensuche in G gilt dann $v.d < u.f$. **1 Punkt**

c) Geben Sie ein Beispiel für eine Tiefensuche in einem gerichteten Graphen G an, in der ein Baum mit einem einzelnen Knoten u gebildet wird, obwohl u sowohl eingehende als auch ausgehende Kanten hat. **2 Punkte**

Aufgabe 3 – Bäume transformieren

Als ADS-HörerIn stellen Sie sich zu Hause natürlich nicht irgendeine Zimmerpflanze auf, sondern einen binären Suchbaum. Ihr Suchbaum heißt B_1 und hat n Knoten. Ihr(e) Übungspartner(in) hat auch einen binären Suchbaum, der B_2 genannt wird und zufälligerweise genau dieselben Schlüssel wie Ihr Suchbaum enthält. Er sieht aber schöner aus. Deshalb möchten Sie Ihren Suchbaum so transformieren, dass er am Ende wie B_2 aussieht.

Zeigen Sie hierzu, dass Sie stets in der Lage sind B_1 durch $O(n)$ Rotationen in B_2 zu transformieren, egal wie die beiden binären Suchbäume konkret aussehen.

Hinweis: Zeigen Sie zunächst, dass $n - 1$ Rotationen ausreichen, um einen beliebigen Suchbaum in einen Suchbaum zu transformieren, in dem kein Knoten einen linken Kindknoten hat. **5 Punkte**

Bitte werfen Sie Ihre Lösungen bis **Donnerstag, 16. Januar 2020, 13:00 Uhr** in den Vorlesungs-Briefkasten im Informatik-Gebäude. Geben Sie stets die Namen und Übungsgruppen aller BearbeiterInnen sowie die Übungsgruppe, in der das Blatt zurückgegeben werden soll, an.

Grundsätzlich sind stets alle Ihrer Aussagen zu begründen und Ihr Pseudocode ist stets zu kommentieren.

Die Lösungen zu den mit PABS gekennzeichneten Aufgaben, geben Sie bitte nur über das PABS-System ab. Vermerken Sie auf Ihrem Übungsblatt, in welchem Repository

(sXXXXXX-Nummer) die Abgabe zu finden ist. Geben Sie Ihre Namen hier als Kommentare in den Quelltextdateien an.