RWTH Aachen
Lehrgebiet Theoretische Informatik
Rossmanith–Hensel–Kuinke–Sánchez

SS 2016
Lösungsvorschlag 9
20. Juni 2016

Übung zur Vorlesung Datenstrukturen und Algorithmen

Aufgabe T26

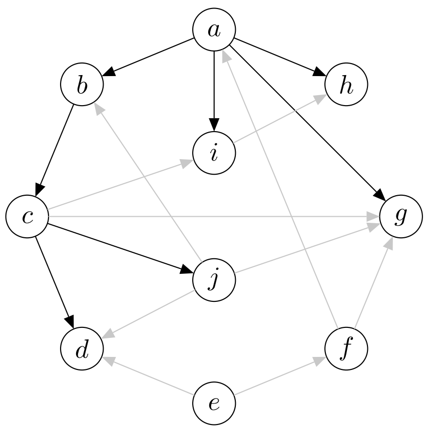
Zunächst können wir die starken Zusammenhangskomponenten in der gewünschten Zeit ermitteln. Ein Knoten liegt auf keinem Kreis genau dann wenn er keine Schleife hat und er in einer starken Zusammenhangskomponenten der Größe eins liegt.

Aufgabe T27

Diese Aussage ist wahr.

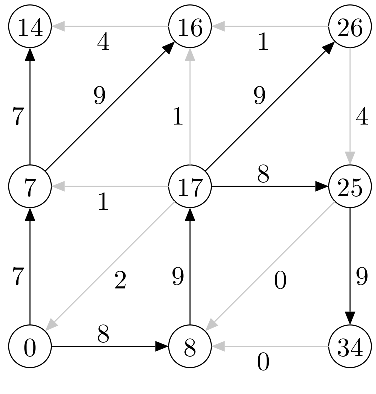
Falls jeder Knoten unterschiedlichen Grad hat, und es n Knoten gibt, müssen alle Knotengrade von 0 bis $n-1$ vorkommen, da der maximale Grad höchstens $n-1$ sein kann. Der Knoten mit Grad $n-1$ kann nicht mit dem Knoten mit Grad 0 verbunden sein. Er hat also nur $n-2$ mögliche Nachbarn. Also kann er nicht $n-1$ viele Kanten haben. Damit haben wir einen Widerspruch, und gezeigt, daß die Aussage wahr ist.

Aufgabe T28



Die Knoten e und f sind nicht erreichbar, alle anderen schon. Die kürzesten Wege finden wir, wenn wir die Baumkanten vom Zielknoten rückwärts verfolgen.

Aufgabe T29



Aufgabe H23
Die Aussage ist wahr: Wenn es in G' eine Vorwärtskante (u, v) gibt, dann gibt es auch einen Pfad aus Baumkanten von u nach v, der mindestens zwei Kanten lang ist. In G bilden dieser Pfad und die Kante $\{u, v\}$ einen Kreis (mindestens der Länge drei).

Nehmen wir umgekehrt an, es gibt einen Kreis in G. Dann gibt es auch einen gerichteten Kreis in G'. Sei v der Kreisknoten mit der kleinsten discovery time. Dann ist die Kreiskante e, die auf v zurückführt, eine Rückwärtskante: Alle Knoten auf dem Kreis liegen in der gleichen starken Zusammenhangskomponente, so daß e keine Querkante sein kann. Sie kann auch keine Baumkante sein, da v von einem anderen Knoten aus zuerst erreicht wurde. Schließlich kann e keine Vorwärtskante sein, weil die discovery time ihres Startknotens größer ist als die von v. Der Rückkante von e steht natürlich eine Vorwärtskante gegenüber.

Aufgabe H24
Wir verwenden Breitensuche:

```
public class H24 {
  static Map<Integer, Integer> dist;
  static Map<Integer, Integer> parent;
  static Queue<Integer> active;

  static void insert(Integer n, Integer m) {
    if(m ≥ 0 &&!dist.containsKey(m)) {
      dist.put(m, dist.get(n) + 1);
      active.add(m);
      parent.put(m, n);
    }
  }

  public static void main(String args[]) {
    dist = new Hashtable<Integer, Integer>();
    parent = new Hashtable<Integer, Integer>();
    active = new LinkedList<Integer>();
    dist.put(1, 0);
    active.add(1);
    while(!active.isEmpty()) {
      int n = active.remove();
      insert(n, n * 3);
      insert(n, n * 2 − 323);
      insert(n, n + 27);
      insert(n, n − 13);
      if(n == 10000) {
        while(n ≠ 1) {
          System.out.println(n);
          n = parent.get(n);
        }
        System.out.println(1);
        return;
      }
    }
  }
}
```

Die Ausgabe des Programms ist:

```
10000
10013
10026
9999
3333
1111
717
239
252
84
28
1
```

Aufgabe H25

Wir wissen aus der Vorlesung, daß

$$C_n \leq n + 1 + \frac{2}{n} \sum_{i=\lceil n/2 \rceil}^{n-1} C_i$$

gilt. Der Induktionsanfang ist durch $C_0 = C_1 = 0$ gegeben. Beweisen wir also jetzt, daß $C_n \leq 4n$ für alle n gilt. Im Beweis werden wir die Gaußsche Summenformel benutzen:

$$\sum_{i=1}^{n} i = \frac{n(n+1)}{2} \qquad (*)$$

Gehen wir nun davon aus, daß $C_i \leq 4i$ für alle $i < n$ bereits erfüllt ist.

$$\begin{aligned}
C_n &\leq n + 1 + \frac{2}{n} \sum_{i=\lceil n/2 \rceil}^{n-1} C_i \\
&\overset{\text{I.V.}}{\leq} n + 1 + \frac{2}{n} \sum_{i=\lceil n/2 \rceil}^{n-1} 4i \\
&\leq n + 1 + \frac{8}{n} \sum_{i=\lceil n/2 \rceil}^{n-1} i \\
&\leq n + 1 + \frac{8}{n} \Big(\sum_{i=1}^{n-1} i - \sum_{i=1}^{\lceil n/2 \rceil - 1} i \Big) \\
&\overset{(*)}{\leq} n + 1 + \frac{8}{n} \Big(\frac{(n-1)n}{2} - \frac{(\lceil \frac{n}{2} \rceil - 1)\lceil \frac{n}{2} \rceil}{2} \Big) \\
&\leq n + 1 + \frac{8}{n} \Big(\frac{(n-1)n}{2} - \frac{(\frac{n}{2} - 1)\frac{n}{2}}{2} \Big) \\
&\leq n + 1 + \frac{8}{n} \Big(\frac{n^2 - n}{2} - \frac{n^2 - 2n}{8} \Big) \\
&\leq n + 1 + \frac{8}{n} \Big(\frac{4n^2 - 4n}{8} - \frac{n^2 - 2n}{8} \Big) \\
&\leq n + 1 + \frac{8}{n} \Big(\frac{4n^2 - 4n - n^2 + 2n}{8} \Big) \\
&\leq n + 1 + \frac{8}{n} \Big(\frac{3n^2 - 2n}{8} \Big) \\
&\leq n + 1 + 3n - 2 \\
&\leq 4n - 1 \\
&\leq 4n
\end{aligned}$$