Lehrstuhl für Informatik I
Algorithmen und Komplexität
Universität Würzburg

Würzburg, den 19. Januar 2023

Prof. Dr. Alexander Wolff
Felix Klesen, M. Sc.

9. Übungsblatt zur Vorlesung Algorithmen und Datenstrukturen (Winter 2022/23)

Aufgabe 1 – Binär hochzählen

Die Datenstruktur D enthält eine einzige natürliche Zahl Z in Binärdarstellung (zu Beginn ist $Z = 0$) und stellt lediglich die Methode Increment zur Verfügung, die Z um Eins erhöht. Die Laufzeit der Methode entspricht dabei der Anzahl der Bits, die sich in Z durch die Erhöhung um Eins ändern.

Zeigen Sie, dass die *amortisierte* Laufzeit von Increment in $O(1)$ ist! **4 Punkte**

Aufgabe 2 – Schlange aus zwei Stapeln analysieren

Zwei Stapel S_1, S_2 können wie folgt eine Schlange Q simulieren:

- Enqueue(x): führe S_1.Push(x) aus – lege also x auf den ersten Stapel.
- Dequeue(): Ist S_2 leer, führe solange S_2.Push(S_1.Pop()) aus, bis S_1 leer ist – verschiebe die Elemente der Reihe nach von S_1 auf S_2. Danach führe S_2.Pop() aus.
- Empty(): gibt true zurück, falls S_1 und S_2 leer sind, sonst false.

Betrachten Sie nun eine zufällige Folge der Länge n bestehend aus Enqueue-, Dequeue- und Empty-Operationen auf Q.

a) Argumentieren Sie, warum die *Worst-Case*-Laufzeit einer einzelnen Dequeue-Operation auf Q in $\Theta(n)$ liegt. **3 Punkte**

b) Zeigen Sie mit amortisierter Analyse: die Gesamtlaufzeit für jede Folge liegt ebenfalls in $\Theta(n)$. Eine einzelne Dequeue-Operation benötigt amortisiert also nur konstante Laufzeit. Wieso ist dies kein Widerspruch zu Teilaufgabe a)? **3 Punkte**

Aufgabe 3 – Zweifärbbarkeit

Ein Graph $G = (V, E)$ heißt *zweifärbbar*, wenn eine Abbildung $c\colon V \to \{\text{rot}, \text{blau}\}$ existiert, so dass für jede Kante $\{u, v\} \in E$ gilt, dass $c(u) \neq c(v)$. Zwei zueinander benachbarte Knoten erhalten also stets unterschiedliche Farben.

a) Welches ist der kleinste Graph, der nicht zweifärbbar ist? **1 Punkt**

b) Entwerfen Sie einen Algorithmus in Pseudocode, der für einen gegebenen Graphen $G = (V, E)$ ermittelt, ob er zweifärbbar ist. Die Laufzeit des Algorithmus soll $O(|V| + |E|)$ sein. Achtung: G ist nicht notwendigerweise zusammenhängend. **3 Punkte**

Aufgabe 4 – Springer auf Schachfeld

Gesucht ist ein Algorithmus, der ermittelt, wieviele Züge ein Springer benötigt, um auf einem Schachbrett mit $n \times n$ Feldern von Feld (x_1, y_1) zu Feld (x_2, y_2) zu kommen.

a) Wie lässt sich dieses Problem als Kürzeste-Wege-Problem auf einem Graphen modellieren? Was repräsentieren die Knoten und Kanten des Graphen? **1 Punkt**

b) Ist der Graph zweifärbbar (entsprechend der Definition in Aufgabe 3)? Begründen Sie Ihre Antwort. **1 Punkt**

c) Geben Sie für das gewöhnliche Schachbrett mit 8×8 Feldern die genaue Anzahl der Kanten des Graphen an. **1 Punkt**

d) Geben Sie für ein Schachbrett mit $n \times n$ Feldern eine obere Schranke für die Anzahl der Kanten an. Verwenden Sie dafür die Groß-O-Notation. **1 Punkt**

e) Geben Sie einen Algorithmus an, der die erforderliche Anzahl an Zügen in $\Theta(n^2)$ Zeit berechnet. Begründen Sie die Laufzeit Ihres Algorithmus. **2 Punkte**

Bitte geben Sie Ihre Lösungen bis **Donnerstag, 26. Januar 2023, 14:00 Uhr** einmal pro Gruppe über Wuecampus als pdf-Datei ab. Vermerken Sie dabei stets die Namen und Übungsgruppen aller BearbeiterInnen auf der Abgabe.

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.