ETH Zürich
Institut für Theoretische Informatik
Prof. Dr. Angelika Steger
Ralph Keusch, Florian Meier

HS 2016
Abgabe 22.11.2016

Algorithmen und Komplexität
Übungsblatt 9

Aufgabe 1

In eine *Queue* (Warteschlange) kann mit der Operation ENQUEUE(x) ein Objekt x eingefügt werden. Die Operation DEQUEUE() gibt jenes Objekt y zurück, das unter allen Objekten in der Queue als erstes mit ENQUEUE(y) eingefügt wurde. Nach der Rückgabe wird y aus der Queue gelöscht.

In einen *Stack* (Stapel) kann mit der Operation PUSH(x) ein Objekt eingefügt werden. Die Operation POP(x) gibt jenes Objekt y zurück, das unter allen Objekten im Stack als letztes mit PUSH(y) eingefügt wurde. Nach der Rückgabe wird y aus dem Stack gelöscht.

Verwenden Sie zwei Stacks um eine Queue zu implementieren, so dass die amortisierte Laufzeit der Operationen ENQUEUE und DEQUEUE jeweils $\mathcal{O}(1)$ ist. Sie können dabei annehmen, dass die Operationen POP und PUSH für einen Stack genau einen Zeitschritt kosten.

Aufgabe 2

Wir betrachten Union-Find-Strukturen mit Path Compression und implementieren die `Union`-Operation wie im Skript beschrieben, d.h. der Baum mit geringerer Höhe wird an den Baum mit grösserer Höhe angehängt. Wir nehmen an, dass insgesamt k Operationen auf einer solchen anfangs leeren Union-Find-Struktur ausgeführt werden, wobei alle `Find`-Operationen am Schluss (d.h. nach allen `MakeNewSet`- und `Union`-Operationen) ausgeführt werden. Zeigen Sie, dass dafür insgesamt nur $\mathcal{O}(k)$ Laufzeit benötigt wird.

Hinweis: Nehmen Sie an, dass sich die k Operationen in eine erste Phase von k_1 `MakeNewSet`- oder `Union`-Operationen und eine zweite Phase von $k_2 = k - k_1$ `Find`-Operationen aufteilen. $\phi(i)$ sei die Anzahl aller Knoten, die von ihrer Wurzel nach der i-ten Operation Abstand mindestens 2 haben. Zeigen Sie, dass die Gesamt-Laufzeit für die k_2 `Find`-Operationen höchstens $\mathcal{O}(\phi(k_1) + k_2)$ ist, und dass man daraus folgern kann, dass die Gesamt-Laufzeit aller k Operationen $\mathcal{O}(k)$ ist.

Aufgabe 3

Wir betrachten *Doppeltes Hashen* zur Auflösung von Kollisionen (siehe Skript). Dazu benutzen wir die zusammengesetzte Hash-Funktion

$$h(k,i) = (h_1(k) + ih_2(k)) \mod m.$$

Zeigen Sie, dass genau ein d-tel aller Speicherplätze für den Schlüssel k in Frage kommen, wobei d den grössten gemeinsamen Teiler von $h_2(k)$ und m bezeichnet.

ABGABE DER HAUSAUFGABEN IN DER VORLESUNG AM 22.11.2016.