RWTH Aachen **SS 2011**
Lehrgebiet Theoretische Informatik Übungsblatt 8
Emmes—Langer—Reidl—Rossmanith 30.05.2011

Übung zur Vorlesung Datenstrukturen und Algorithmen

Aufgabe T25

	Quicksort	Heapsort	Mergesort	Insertion-Sort	Straight-Radix	Radix-Exchange
in-place?						
stabil?						
Laufzeit (worst-case)						
Laufzeit (Durchschnitt)						
vergleichsbasiert?						

Beantworten Sie die Fragen für alle Sortierverfahren. Gehen Sie davon aus, daß ein Vergleich in konstanter Zeit durchgeführt wird und die Anzahl der zu sortierenden Elemente n beträgt. Für Laufzeiten tragen Sie eine Funktion $f(n)$ in die Tabelle ein, um eine Laufzeit von $O(f(n))$ auszudrücken.

Aufgabe T26

	Liste	Sortierte Liste	Array	Sortiertes Array	Trie	Binärer Suchbaum	AVL-Baum	Splay-Tree	Treap	Skip-List	Hashtabelle	Min-Heap
Randomisiert?												
Einfügen												
Suchen												
Löschen												
Minimum												
Maximum												
Sort. Ausgeben												

Beantworten Sie die Fragen für alle Datenstrukturen bzw. geben Sie eine obere Schranke für den Aufwand an. Gehen Sie davon aus, daß die Anzahl der enthaltenden Elemente n beträgt. Für Laufzeiten tragen Sie eine Funktion $f(n)$ in die Tabelle ein, um eine Laufzeit von $O(f(n))$ auszudrücken. Bei randomisierten Datenstrukturen ist die erwartete obere Schranke gemeint, bei amortisierten Analysen die amortisierte Laufzeit.

Aufgabe T27

Entwickeln Sie eine Möglichkeit, die Adjazenzmatrix eines ungerichteten Graphens mit n Knoten in höchstens $n(n-1)/2$ Einträgen zu speichern, so daß es trotzdem in $O(1)$ Schritten möglich ist zu entscheiden, ob zwei Knoten i, j adjazent sind.

Aufgabe H20 (10 Punkte)

Betrachten Sie wieder die Darstellung der Adjazenzmatrix eines ungerichteten Graphens mit n Knoten in einem Array aus Tutoraufgabe T26. Wir möchten basierend auf dieser Darstellung nun möglichst effizient die Nachbarschaft eines Knotens bestimmen, d.h. die Menge $N(j)$ der Knoten, zu denen der Knoten j adjazent ist. Beispielsweise möchte man oft wissen, wie viele Nachbarn der Knoten j hat (dieses nennt man den Grad des Knotens j).

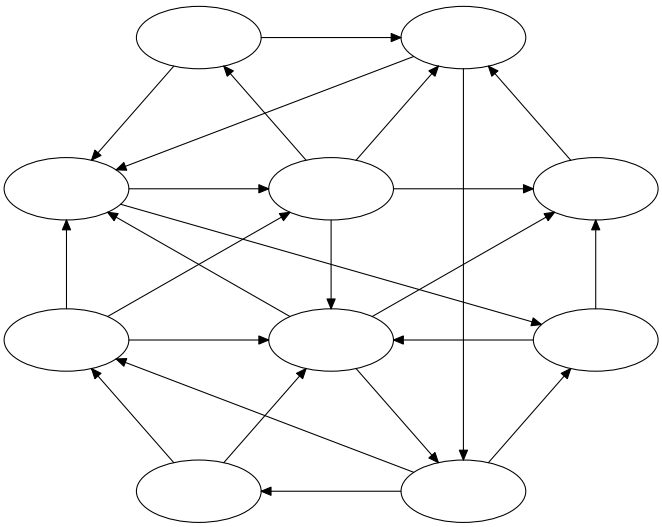
Entwickeln Sie ein Verfahren, welches anhand des o.g. Arrays die Nachbarschaft eines Knotens in $\Theta(n)$ Schritten berechnet, und dabei nur eine *konstante* Anzahl an Multiplikationen verwendet.

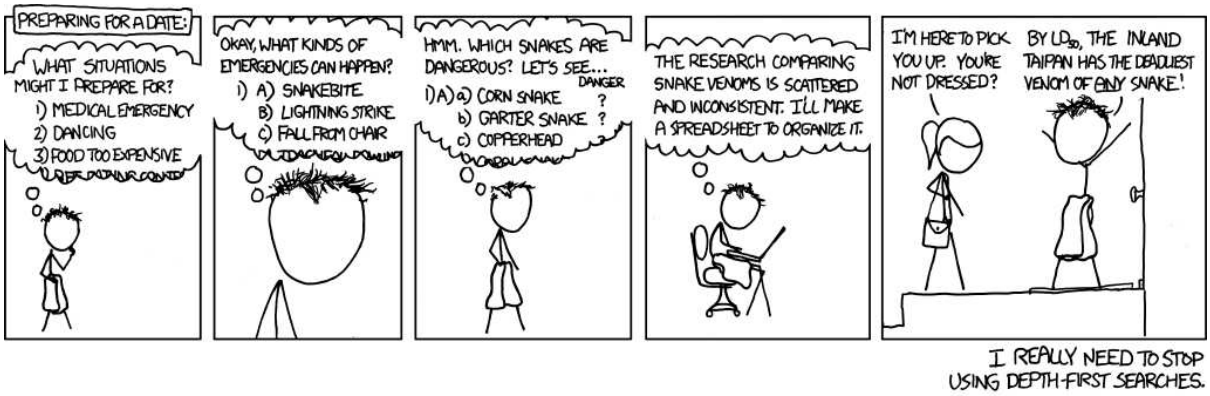
Aufgabe H21 (10 Punkte)

Beweisen oder widerlegen Sie die folgende Aussage: Wenn eine Tiefensuche auf einem ungerichteten Graphen durchgeführt wird, erhalten wir keine Querkanten.

Aufgabe H22 (10 Punkte)

Führen Sie eine Tiefensuche auf dem folgenden Graphen durch. Geben Sie zu jedem Knoten *discovery* und *finish* Zeiten an, und geben Sie dann zu jeder Kante an, welchen Typ sie bezüglich des entstandenen Tiefensuchwaldes hat (Baumkante, Vorwärtskante, Rückwärtskante, Querkante).





(Erlaubter Abdruck von der Webseite `http://xkcd.com/`)