

Übungen zur Vorlesung

Grundlagen der Programmierung II

Blatt 6

Aufgabe 1:

- Schreiben Sie eine Funktion (am besten in PASCAL), die einen Graph $G=(V,E)$ mit $V=\{1,\dots,n\}$ einliest und als Ergebnis einen Zeiger auf die Adjazenzlistendarstellung des Graphen liefert. Die Eingabe bestehe aus der Zahl n gefolgt von den Kanten $\{i,j\}\in V$ in Form von Paaren $i;j$.
- Schreiben Sie eine Prozedur zum Löschen eines Knoten aus dem Graphen.
- Der **Abstand** zweier Knoten u, v ist definiert als die Länge des kürzesten Weges zwischen u und v . Schreiben Sie eine Funktion, die zu einem beliebigen Knoten v des Graphen eine Liste aller Knoten im Abstand 2 von v liefert.

Aufgabe 2:

- Schreiben Sie einen Algorithmus, der aus der Binärdarstellung einer natürlichen Zahl n die Zahl 2^n berechnet.
Die Eingabe des Algorithmus ist die natürliche Zahl k und die Folge $b_k\dots b_0$ mit $b_i\in\{0,1\}$ $\forall i$ und $b_k=1$. (die Binärdarstellung einer natürlichen Zahl)
Der Algorithmus soll im Einheitskostenmodell höchstens $2^{\lceil \log n \rceil} + 1$ Multiplikationen benötigen. Dabei dürfen Multiplikationen nicht durch Folgen von Additionen oder Shift-Operationen ersetzt werden.
- Für welche $k+1$ Bit Eingabefolge benötigt der konstruierte Algorithmus die meisten Multiplikationen?
- Wenn man die realistische Annahme macht, daß eine Multiplikation $a*b$ Aufwand in Höhe von $\lceil \log a \rceil + 1 + \lceil \log b \rceil + 1$ im logarithmischen Kostenmodell verursacht, wie groß ist dann die Zeitkomplexität des in (a) konstruierten Algorithmus? Führen Sie einen exakten Beweis!

Anm. [...] steht für die Pascal-Operation $\text{trunc}()$, d.h. die Umwandlung eines real-Wertes in einen integer-Wert durch Abschneiden der Nachkommastellen.

Aufgabe 3:

Beweisen Sie:

- Der Algorithmus A mit $T_A(n)=n^k$ ist asymptotisch schneller als der Algorithmus B mit $T_B(n)=2^n$ für alle $n,k\in\mathbb{N}$.
- Für zwei Funktionen $f,g: \mathbb{N}\rightarrow\mathbb{N}$ mit $f(n)>0$ und $g(n)>0$ für alle $n\in\mathbb{N}$ gilt:
$$\max\{f(n),g(n)\}=O(f(n)+g(n))$$
- Für zwei Funktionen $f,g: \mathbb{N}\rightarrow\mathbb{N}$ mit $f(n)>0$ und $g(n)>0$ und für alle $n\in\mathbb{N}$ gilt:
$$f(n)+g(n)=O(\max\{f(n),g(n)\})$$

Aufgabe 4:

Bestimmen Sie nach dem in der Vorlesung angegebenen Verfahren den binären Baum $B(F)$, der folgenden Wald repräsentiert:

