

Jetzt wollen wir noch etwas mehr: den direkten Zugriff auf die Nachbarn der Knoten, wenn wir sie brauchen!

Also:



Zugriff auf Nachbarn jeweils ab Semikolon.

Algorithmisch: Gehe Liste durch, zähle Semikolons → das dauert

Datenstruktur: Speichere die Stelle ab, an der die Nachbarn von v_3 zu finden sind.

→ Zeiger (engl. "Pointer")

Bekannt bei Webseiten:

Speicherinhalt : z.B. YouTube-Video (etliche MB)

Zeiger : URL (einige Byte)

„Man muss nicht alles wissen, man muss nur wissen wo's steht!“

Für den direkten Zugriff brauchen wir n Zeiger; jeder codiert eine Speicherzelle, d.h. die Nummer eines Bits in der Liste.

So ein Zeiger braucht also selber

$$\log_2 (2n + 4m + n(\log_2 n + 1) + 2m(\log_2 n + 1))$$

für $n \geq 10$
 $m \geq n$

$$\leq \log_2 (9m(\log_2 n + 1) + 1)$$

$$\leq \log_2 9 + \log_2 m + \log_2 (\log_2 n + 1) + 1 \leq 2 \log_2 m$$

Bits

Insgesamt benötigen wir also $\Theta(n \log_2 m)$ Bits.

Setzt ist $m \leq n^2$,

also $\log_2 m \leq \log_2 n^2 = 2 \log_2 n$,

d.h. $n \log m \leq 2n \log n$,

und der insgesamt verbrauchte Speicherplatz ist

$$\Theta(n \log m + m \log n) = \Theta(m \log n).$$

3.7 WACHSTUM VON FUNKTIONEN

Im letzten Abschnitt haben wir Funktionen abgeschätzt und auf die "wesentlichen" Bestandteile reduziert, um Größenordnungen und Wachstumsverhalten zu beschreiben. Ein bisschen formaler:

DEFINITION 3.9 (Θ -Notation)

Seien $f, g: \mathbb{N} \rightarrow \mathbb{R}$ Funktionen.

Dann gilt

$$f \in \Theta(g) \Leftrightarrow \text{Es gibt positive Konstanten } c_1, c_2, n_0 \text{ mit } 0 \leq c_1 g(n) \leq f(n) \leq c_2 g(n) \text{ für alle } n \geq n_0.$$

Man sagt: f wächst asymptotisch in derselben Größenordnung wie g .

Beispiele: $2n^2 - 1 \in \Theta(n^2)$

$$\frac{n^3}{1000} + n^2 + n \log n \in \Theta(n^3)$$

Manchmal hat man nur untere oder obere Abschätzungen:

DEFINITION 3.10 (O -Notation)

Seien $f, g: \mathbb{N} \rightarrow \mathbb{R}$ Funktionen.

Dann gilt $f \in O(g) \Leftrightarrow$ Es gibt positive Konstanten c, n_0 mit
 $0 \leq f(n) \leq c g(n)$ für alle $n \geq n_0$.

Man sagt: f wächst ~~exponentiell~~ höchstens in derselben Größenordnung wie g .

Beispiele:
 $2n^2 - 1 \in O(n^2)$
 $2n^2 - 1 \in O(n^3)$
 $n \log n \in O(n^2)$

DEFINITION 3.11 (Ω -Notation)

Seien $f, g: \mathbb{N} \rightarrow \mathbb{R}$ Funktionen.

Dann gilt $f \in \Omega(g) \Leftrightarrow$ Es gibt positive Konstanten c, n_0 mit
 $0 \leq c g(n) \leq f(n)$ für alle $n \geq n_0$.

Beispiele: $2n^2 - 1 \in \Omega(n^2)$
 $2n^2 - 1 \in \Omega(n^3)$

Einige einfache Eigenschaften:

SATZ 3.12

Seien $f, g: \mathbb{N} \rightarrow \mathbb{R}$ Funktionen.

- Dann gilt
- (1) $f \in O(g) \Leftrightarrow g \in O(f)$
 - (2) $f \in O(g) \Leftrightarrow f \in O(g)$ und $f \in \Omega(g)$.
 - (3) $f \in O(g) \Leftrightarrow g \in \Omega(f)$

Beweis: Übung!