Algoritmid ja andmestruktuurid

(MTAT.03.133, 4 AP)

Loengud:	E 10:15, aud. 111
Praktikumid:	T 14:15, aud. 111

koduleht:

`http://www.ut.ee/~peeter_l/teaching/a_ja_a03s`

(sinna kogunevad loengukiled)

Hinde saamiseks: praktikumiarvestus ja eksam.

Kirjandus:

Jüri Kiho. *Algoritmid ja andmestruktuurid (kolmas trükk)*. TÜ 2003.

Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest. *Introduction to Algorithms*. MIT Press, 1990.

Algoritm on arvutussammude tegemise eeskiri, et mingitest sisenditest välja arvutada mingid väljundid.

Käesolevas loengus uurime, kuidas hinnata algoritmi tehtavate sammude arvu sõltuvalt sisendi suurusest.

Antud algoritmi jaoks huvitab meid funktsioon f, mille korral $f(n)$ on selle algoritmi tehtavate sammude maksimaalne arv sisenditel, mille suurus on n. („ajaline keerukus halvimal juhul") Mõnikord huvitab ka

- keskmine keerukus;
- amortiseeritud keerukus.

Näide: rea summa.

Sisend: Arvumassiiv a. Suurus n on massiivi pikkus.

Algoritm:

1	$s := 0$	täidetakse 1 kord
2	**for** $i := 1$ **to** n	täidetakse n korda
3	$\quad s := s + a_i$	täidetakse n korda
4	**return** s	täidetakse 1 kord

Kokku tehakse $(c_2 + c_3)n + c_1 + c_4$ sammu, kus c_i on i-ndal real tehtavate sammude arv.

Näide: mullisort.

Sisend: Massiiv a. Suurus n on massiivi pikkus.

Algoritm:

for $i := 1$ **to** $n-1$	$n-1$ korda
for $j := i+1$ **to** n	$n(n-1)/2$ korda
if $a_i > a_j$	$n(n-1)/2$ **then** korda
$a_i :=: a_j$	ülimalt $n(n-1)/2$ korda
return a	1 kord

Kokku tehakse ülimalt $\frac{c_2+c_3+c_4}{2}n^2 + \frac{2c_1-c_2-c_3-c_4}{2}n + (c_5-c_1)$ sammu ning vähemalt $\frac{c_2+c_3}{2}n^2 + \frac{2c_1-c_2-c_3}{2}n + (c_5-c_1)$ sammu.

Rea summeerimisel tehakse ülimalt $pn + q$ sammu mingite konstantide p ja q korral ning mullisordil ülimalt $pn^2 + qn + r$ sammu mingite konstantide p, q ja r korral.

Konstantide täpsed väärtused sõltuvad realisatsioonist, me ei püüa neid leida.

Kui n on küllalt suur, siis pn on enam-vähem sama suur kui $pn+q$ ning pn^2 on enam-vähem sama suur kui pn^2+qn+r.

Olgu f ja g kaks naturaalarvuliste argumentidega ja positiivsete väärtustega funktsiooni.

f on $O(g)$, kui leiduvad $c > 0$ ja $n_0 \in \mathbb{N}$ nii, et iga $n \geqslant n_0$ korral $f(n) \leqslant cg(n)$.

$pn + q$ on $O(n)$ ja $pn^2 + qn + r$ on $O(n^2)$. Kordajaks c võib võtta suvalise arvu, mis on suurem kui p.

Lause. f on $O(g)$ parajasti siis, kui leidub $c > 0$ nii, et iga $n \in \mathbb{N}$ korral $f(n) \leqslant cg(n)$.

f on $\Theta(g)$, kui f on $O(g)$ ja g on $O(f)$, s.t. leidub $c > 0$ ja $n_0 \in \mathbb{N}$ nii, et iga $n \geqslant n_0$ korral $\frac{g(n)}{c} \leqslant f(n) \leqslant cg(n)$.

f on $\Omega(g)$, kui g on $O(f)$.

Kui sama ülesande lahendamiseks on teada kaks erinevat algoritmi, millest ühe keerukushinnang on parem (väiksem) kui teise oma, siis on tõenäoline, et vähegi suuremate sisendite korral töötab esimene algoritm kiiremini.

- $c \cdot f$ on $O(f)$.
- Kui f on $O(h)$ ja g on $O(h)$, siis $f + g$ on $O(h)$.
- Kui f on $O(g)$ ja g on $O(h)$, siis f on $O(h)$.
- Kui $r \leqslant s$, siis n^r on $O(n^s)$.
- k-nda astme polünoom on $O(n^k)$.
- Kui f_1 on $O(g_1)$ ja f_2 on $O(g_2)$, siis $f_1 f_2$ on $O(g_1 g_2)$.
- n^k on $O(c^n)$, kui $c > 1$.
- $\log_b n$ on $O(n^k)$, kui $b > 1$ ja $n > 0$.
- $\log_b n$ on $\Theta(\log_d n)$ iga $b, d > 1$ korral.
- $\sum_{i=1}^{n} i^r$ on $\Theta(n^{r+1})$.

Kui f on funktsioon, siis kirjutis $O(f)$ tähistab ka kõigi selliste funktsioonide g hulka, mis on $O(f)$. $\Theta(f)$ ja $\Omega(f)$ omavad sarnaseid tähendusi.

Kirjutis $A(O(f_1), \ldots, O(f_k)) = A'(O(f'_1), \ldots, O(f'_l))$, kus A ja A' on mingid avaldised ning $f_1, \ldots, f_k, f'_1, \ldots, f'_l$ mingid funktsioonid tähendab, et
iga $g_1 \in O(f_1), \ldots, g_k \in O(f_k)$ jaoks
leiduvad $g'_1 \in O(f'_1), \ldots, g'_l \in O(f'_l)$ nii, et
$A(g_1, \ldots, g_k) = A'(g'_1, \ldots, g'_l)$.

Näiteks kirjutis $f(n) = O(n^2)$ tähendab, et $f(n)$ on $O(n^2)$.

Kui f on $O(g)$, siis $f(n) = O(1)g(n)$.

Näide: lausearvutusvalemi rahuldatavus.

Sisend: Lausearvutusvalem $\mathcal{A}$ (koosneb muutujatest ja loogilistest tehetest). Suuruseks n loeme muutujate arvu.

Algoritm: rahuldatav?($\mathcal{A}$), kus rahuldatav?($\mathcal{A}$) on

„lihtsusta“ $\mathcal{A}$
if $\mathcal{A} \equiv \mathsf{true}$ **then**
    return TRUE
else if $\mathcal{A} \equiv \mathsf{false}$ **then**
    return FALSE
Olgu x üks $\mathcal{A}$ muutujatest
$\mathcal{A}_t := \mathcal{A}_{x \leftarrow \mathsf{true}}$; $\mathcal{A}_f := \mathcal{A}_{x \leftarrow \mathsf{false}}$
return rahuldatav?($\mathcal{A}_t$) $\vee$ rahuldatav?($\mathcal{A}_f$)

Olgu $T(n)$ programmi tööaeg (halvimal juhul) sisendil suurusega n. Siis $T(1) = c$ ja $T(n) \leqslant 2T(n-1) + p(n)$ mingi konstandi c ja polünoomi p jaoks.

[Pole päris korrektne, aga loeme siin, et valemi pikkus on polünomiaalne muutujate arvu suhtes.]

Seega $T(n) \leqslant 2^{n-1}c + \sum_{i=2}^{n} 2^{n-i}p(i) = O(np(n)2^n) = 2^{O(n)}$.

Näide: kahendotsimine.

Sisend: Sorteeritud massiiv a ja element x. Suurus n on massiivi pikkus.

Algoritm: otsi$(a, x, 1, n)$, kus otsi(a, x, l, r) on

if $l > r$ **then**
    return FALSE
$k := \lfloor (l + r)/2 \rfloor$
if $a_k = x$ **then**
    return TRUE
else if $x < a_k$ **then**
    return otsi$(a, x, l, k - 1)$
else
    return otsi$(a, x, k + 1, r)$

Olgu $T(n)$ programmi tööaeg (halvimal juhul) sisendil suurusega n. Siis $T(1) = c$ ja $T(n) \leqslant T(n/2) + c'$ mingite konstantide c, c' jaoks.

$T(n) \leqslant \underbrace{c' + c' + \cdots + c'}_{\log_2 n \text{ korda}} + c = c + c' \log_2 n$. Seega $T(n)$ on $O(\log n)$.

Teoreem. („põhiteoreem“, “*Master Theorem*”) Olgu $a \geqslant 1$, $b > 1$, olgu $f(n)$ mingi kasvav funktsioon. Tähistame $k = \log_b a$. Olgu $T(n)$ defineeritud järgmiselt:

$$T(1) = t$$
$$T(n) = a \cdot T(n/b) + f(n) \quad .$$

Siin n/b võib olla kas $\lfloor n/b \rfloor$ või $\lceil n/b \rceil$.

- Kui $f(n)$ on $O(n^{k-\varepsilon})$, siis $T(n)$ on $\Theta(n^k)$.
- Kui $f(n)$ on $\Theta(n^k)$, siis $T(n)$ on $\Theta(n^k \log n)$.
- Kui $f(n)$ on $\Omega(n^{k+\varepsilon})$ ning $af(n/b) \leqslant cf(n)$ mingi $c < 1$ ja kõigi küllalt suurte arvude n jaoks, siis $T(n)$ on $\Theta(f(n))$.

Kahendotsimisel oli $T(n) = T(n/2) + f(n)$, kus $f(n)$ on $\Theta(1)$. Siin $a = 1$, $b = 2$ ja $k = \log_b a = 0$. Seega $f(n)$ on $\Theta(n^k)$. Teoreemi teise variandi järgi siis $T(n)$ on $\Theta(n^k \log n)$ ehk $\Theta(\log n)$.

Tõestus. Kui $n = b^i$, siis

$$\begin{aligned}
T(n) &= f(n) + aT(n/b) = \\
&= f(n) + af(n/b) + a^2T(n/b^2) = \cdots = \\
&= f(n) + af(n/b) + \cdots + a^{i-1}f(n/b^{i-1}) + a^iT(1) = \\
&= T(1)n^k + \sum_{j=0}^{i-1} a^j f(n/b^j) = \Theta(n^k) + \sum_{j=0}^{i-1} a^j f(n/b^j),
\end{aligned}$$

sest $a^i = a^{\log_b n} = n^{\log_b a} = n^k$.

Kirjutades $\Theta(n^k)$ oleme eeldanud, et T on ainult b astmetel defineeritud.

Kui $n = b^i$ ja $f(n) = O(n^{k-\varepsilon})$, siis

$$\sum_{j=0}^{i-1} a^j f(n/b^j) = O\left(\sum_{j=0}^{i-1} a^j((n/b^j)^{k-\varepsilon})\right)$$

$$\sum_{j=0}^{i-1} a^j((n/b^j)^{k-\varepsilon}) = n^{k-\varepsilon}\sum_{j=0}^{i-1}\left(\frac{ab^\epsilon}{b^k}\right)^j =$$

$$= n^{k-\varepsilon}\sum_{j=0}^{i-1} b^{\varepsilon j} = n^{k-\varepsilon}\frac{b^{\varepsilon i}-1}{b^\varepsilon - 1} =$$

$$= n^{k-\varepsilon}\frac{n^\varepsilon - 1}{b^\varepsilon - 1} = O(n^k) \quad .$$

Kui $n = b^i$ ja $f(n) = \Theta(n^k)$, siis

$$\sum_{j=0}^{i-1} a^j f(n/b^j) = \Theta\left(\sum_{j=0}^{i-1} a^j ((n/b^j)^k)\right)$$

$$\begin{aligned}\sum_{j=0}^{i-1} a^j ((n/b^j)^k) &= n^k \sum_{j=0}^{i-1} \left(\frac{a}{b^k}\right)^j = \\ &= n^k \sum_{j=0}^{i-1} 1 = \\ &= n^k \cdot i = n^k \log_b n \quad .\end{aligned}$$

Kui $n = b^i$ ja $af(n/b) \leqslant cf(n)$, kus $c < 1$, siis

$$\sum_{j=0}^{i-1} a^j f(n/b^j) \leqslant \sum_{j=0}^{i-1} c^j f(n) \leqslant$$
$$\leqslant f(n) \sum_{j=0}^{\infty} c^j = f(n) \frac{1}{1-c} = O(f(n)) \ .$$

Kuna see summa sisaldab liiget $f(n)$ ning kõik liikmed on mittenegatiivsed, siis on ta ka $\Omega(f(n))$, kokku seega $\Theta(f(n))$.

Oleme tõestanud teoreemi juhuks, kui $T(n)$ on defineeritud ainult b astmetel.

Olgu n suvaline naturaalarv. Kui $T(n) = aT(\lceil n/b \rceil) + f(n)$, siis on lihtne näidata, et $T(n)$ on $\Omega(\ldots)$ (tingimustel, mille kohta teoreem väidab, et $T(n)$ on $\Theta(\ldots)$). Meil tuleb veel näidata, et $T(n)$ on $O(\ldots)$.

Olgu $n_0, n_1, \ldots$ defineeritud järgmiselt:

$$\begin{aligned} n_0 &= n \\ n_{j+1} &= \lceil n_j/b \rceil \; . \end{aligned}$$

Kuna $n_{j+1} \leqslant \frac{n_j}{b} + 1$, siis $n_j \leqslant \frac{n}{b^j} + \sum_{s=0}^{j-1} \frac{1}{b^s} \leqslant \frac{n}{b^j} + \frac{b}{b-1}$.

Olgu $i = \lfloor \log_b n \rfloor$, siis $n_i = O(1)$.

$$T(n) = a^i T(n_i) + \sum_{j=0}^{i-1} a^j f(n_j) \leqslant \Theta(n^k) + \sum_{j=0}^{i-1} a^j f\left(\frac{n}{b^j} + \frac{b}{b-1}\right)$$

Kui $f(n) = O(n^{k-\varepsilon})$, siis

$$\sum_{j=0}^{i-1} a^j f\left(\frac{n}{b^j} + \frac{b}{b-1}\right) = O\left(\sum_{j=0}^{i-1} a^j \left(\frac{n}{b^j} + \frac{b}{b-1}\right)^{k-\varepsilon}\right)$$

$$\begin{aligned}
\sum_{j=0}^{i-1} a^j \left(\frac{n}{b^j} + \frac{b}{b-1}\right)^{k-\varepsilon} &= \sum_{j=0}^{i-1} a^j \left(\frac{n}{b^j}\right)^{k-\varepsilon} \left(1 + \frac{b^j}{n} \cdot \frac{b}{b-1}\right)^{k-\varepsilon} \leqslant \\
&\leqslant \sum_{j=0}^{i-1} a^j \left(\frac{n}{b^j}\right)^{k-\varepsilon} \left(1 + \frac{b}{b-1}\right)^{k-\varepsilon} = \\
&= O(1) \sum_{j=0}^{i-1} a^j \left(\frac{n}{b^j}\right)^{k-\varepsilon} = O(n^k)
\end{aligned}$$

Kui $f(n) = \Theta(n^k)$, siis analoogiliselt

$$\sum_{j=0}^{i-1} a^j f\left(\frac{n}{b_j} + \frac{b}{b-1}\right) = O(1) \sum_{j=0}^{i-1} a^j \left(\frac{n}{b_j}\right)^k = O(n^k \log n) \ .$$

Kui $af(\lceil n/b \rceil) \leqslant cf(n)$, kus $c < 1$, siis $a^j f(n_j) \leqslant c^j f(n)$ ning

$$\sum_{j=0}^{i-1} a^j f(n_j) \leqslant \sum_{j=0}^{i-1} c^j f(n) = O(f(n)) \ .$$

- Asümptootikuid kasutades tehakse eeldus, et probleemi suurusega võib lõpmatusse minna.
- Arvutiarhitektuuri ebaregulaarsustest tingituna võib mõne algoritmi reaalne tööaeg (arvutile jõukohastel probleemisuurustel) olla sarnasem mõne teise funktsiooniga kui leitud O-hinnang.
- Käesolevas kursuses uurime ikkagi asümptootikuid.

Tartu Akadeemiline Meeskoor võtab vastu uusi lauljaid.

Proovid toimuvad T 18:30 ja N 18:30 endises EPA klubis (Veski 6, Kassitoomel).

Esimene proov toimub 9. septembril.

Eriti oodatud on tenorid.