Algoritmi i strukture podataka
rešeni zadaci 1. deo

Aleksandar Veljković

2017/2018

1 Zadaci

1. Dokazati da važe sledeće jednakosti:

   a) $2n + 3 \in O(n)$

   b) $2n^2 + 3n \in O(n^2)$

   c) $\sqrt{n} - 2\log n \in \Omega(\log n)$

   d) $2^{n-1} + n \in \Theta(2^n)$

2. Ako se algoritam A izvršava na računaru koji izvršava 10^8 instrukcija u sekundi a algoritam A ima funkciju vremenske složenosti koja pripada klasi $O(n^2)$, koja je najveća moguća vrednost veličine ulaznog podatka n za algoritam A da bi algoritam završio sa izvršavanjem za 1 minut?

3. Za koliko sekundi će se izvršiti algoritam A, sa ulazom 1000 vremenske složenosti $O(n^3)$ ako se izvršava na računaru koji izvršava 10^6 instrukcija u sekundi?

4. Koliko je instrukcija u sekundi potrebno da izvršava racunar da bi se na njemu izvršio algoritam složenosti $O(n^7)$ za ulaz veličine $n = 100$ ako je vreme izvršavanja ograničeno na 10 sekundi.

5. Naći opšte rešenje rekurentne jednačine $T(n) = T(n-1) + T(n-2)$ sa početnim uslovima $T(0) = 0$ i $T(1) = 1$. Dobijeno rešenje izraziti u O notaciji.

6. Naći opšte rešenje rekurentne jednačine $T(n) = T(n-1) + n + 1$ sa početnim uslovom $T(1) = 1$. Dobijeno rešenje izraziti u O notaciji.

7. Odrediti broj koraka narednog programskog koda i rešenje izraziti u O notaciji:

```
k = 3;
for(i = 1; i < n; i++)
{
    while(k <= n)
    {
        k = k * 3;
        suma = suma + i;
    }
}
```

2 Rešenja

1. Dokazati da važe sledeće jednakosti:

   a) Da bi važilo $2n + 3 \in O(n)$ dovoljno je pronaći pozitivne konstante c i N tako da važi:

   $$2n + 3 \leq c \cdot n, \ \forall n \geq N$$

   Jedno od mogućih rešenja je: $c = 3$, $N = 3$, dokažimo indukcijom po n da je pronadjeno rešenje ispravno:

   Baza indukcije: $n = 3$

   $$2 \cdot 3 + 3 \leq 3 \cdot 3$$
   $$9 \leq 9$$

   Pošto baza indukcije važi, pretpostavimo da nejednakost važi za neko $n \geq 3$. Treba pokazati da nejednakost važi i za $n + 1$:

   $$2(n + 1) + 3 \leq 3(n + 1)$$
   $$2n + 2 + 3 \leq 3n + 3$$
   $$2n + 3 \leq 3n + 1$$
   $$2n + 3 \leq n + 2n + 1$$

   Kako po hipotezi važi da je $2n+3 \leq n$, važi i nejednakost u kojoj je desnoj strani dodata pozitivna vrednost $2n+1$, pa je time dokazano da je pronadjeno rešenje zaista rešenje nejednakosti a time i da je $2n + 3 \in O(n)$

   b) Da bi važilo $2n^2 + 3n \in O(n^2)$ dovoljno je pronaći pozitivne konstante c i N tako da važi:

   $$2n^2 + 3n \leq c \cdot n^2, \ \forall n \geq N$$

   Podelimo obe strane nejednakosti pozitivnom vrednošću n:

   $$\frac{2n^2}{n} + \frac{3n}{n} \leq \frac{c \cdot n^2}{n}$$
   $$2n + 3 \leq c \cdot n$$

   Jedno od mogućih rešenja je: $c = 3$, $N = 3$, pronadjeno u delu pod a).

   c) Da bi važilo $\sqrt{n} - 2\log n \in \Omega(\log n)$ dovoljno je pronaći pozitivne konstante c i N tako da važi:

   $$\sqrt{n} - 2\log n \geq c \cdot \log n, \ n \geq N$$

   $$\sqrt{n} - 2\log n \geq c \cdot \log n$$
   $$\sqrt{n} \geq c \cdot \log n + 2\log n$$
   $$\sqrt{n} \geq (2 + c) \cdot \log n$$

   Nažalost, u ovom slučaju nije potpuno jasno kako odabrati konstante c i N. Kako važi da n^α raste brže od $(\log n)^\beta$ za $\alpha, \beta > 0$, to navedena nejednakost važi za proizvoljnu konstantu c i dovoljno veliku vrednost N. Da bismo pronašli N potrebno je da znamo osnovu logaritma.

   Ipak, do dokaza da je $\sqrt{n} - 2\log n \in \Omega(\log n)$ možemo doći i nešto drugačijim putem.

   Kako važi da je:

   $$\log n \in \Theta(\ln n)$$

to je polazni problem ekvivalentan problemu:

$$\sqrt{n} - 2\ln n \in \Omega(\ln n)$$

Naizgled, problem deluje podjednako težak kao i polazni. Ipak, ako pokažemo da važi jače tvrdjenje, da je:

$$\sqrt{n} - 2\ln n \in \omega(\ln n)$$

važiće i

$$\sqrt{n} - 2\ln n \in \Omega(\ln n)$$

te i polazno tvrdjenje zadatka.

Da bi važilo:

$$\sqrt{n} - 2\ln n \in \omega(\ln n)$$

potrebno je da je:

$$\lim_{n\to\infty} \frac{\sqrt{n} - 2\ln n}{\ln n} \to \infty$$

Dolazimo do problema da je limes oblika $\frac{\infty}{\infty}$ kada n teži beskonačnosti. Kako su obe funkcije diferencijabilne u svim tačkama n za $1 < N \leq n$, možemo primeniti Lopitalovo pravilo:

$$\lim_{n\to c} \frac{f(x)}{g(x)} = \lim_{n\to c} \frac{f'(x)}{g'(x)}$$

ako su f i g diferencijabilne u tački c, pa za naš limes dobijamo:

$$\lim_{n\to\infty} \frac{\sqrt{n} - 2\ln n}{\ln n} = \lim_{n\to\infty} \frac{\frac{1}{2\sqrt{n}} - \frac{2}{n}}{\frac{1}{n}}$$

$$= \lim_{n\to\infty} \frac{\frac{1}{2\sqrt{n}}}{\frac{1}{n}} - \lim_{n\to\infty} \frac{\frac{2}{n}}{\frac{1}{n}}$$

$$= \lim_{n\to\infty} \frac{1}{2} \cdot \frac{n}{\sqrt{n}} - \lim_{n\to\infty} \frac{2n}{n} = \lim_{n\to\infty} \frac{1}{2} \cdot \sqrt{n} - \lim_{n\to\infty} 2 = \infty - 2 = \infty$$

Time smo pokazali da važi jače tvrdjenje:

$$\sqrt{n} - 2\ln n \in \omega(\ln n)$$

iz čega sledi:

$$\sqrt{n} - 2\ln n \in \Omega(\ln n)$$

a onda i:

$$\sqrt{n} - 2\log n \in \Omega(\log n)$$

Čime je dokazano polazno tvrdjenje.

d) Da bi važilo $2^{n-1} + n \in \Theta(2^n)$ potrebno je da važi:

$$2^{n-1} + n \in O(2^n) \text{ i } 2^{n-1} + n \in \Omega(2^n)$$

ili

$$2^{n-1} + n \in O(2^n) \text{ i } 2^n \in O(2^{n-1} + n)$$

Da bi važilo $2^{n-1} + n \in O(2^n)$ dovoljno je pronaći pozitivne konstante c_1 i N_1 tako da važi:

$$2^{n-1} + n \leq c_1 \cdot 2^n, \ \forall n \geq N_1$$

Jedno rešenje bi moglo biti $c_1 = 1$, $N = 1$

Da bi važilo $2^n \in O(2^{n-1} + n)$ dovoljno je pronaći pozitivne konstante c_1 i N_1 tako da važi:

$$2^n \leq c_2 \cdot 2^{n-1} + n, \ \forall n \geq N_2$$

Jedno zajedničko rešenje za obe nejednakosti bi moglo biti $c_2 = 2$, $N = 1$, pa su $c_1 = 1$, $c_2 = 2$ i $N = max(N_1, N_2) = 1$ pa je time dokazano da važi:

$$2^{n-1} + n \in \Theta(2^n)$$

2. Algoritam vremenske složenosti $O(n^2)$ za jednu sekundu može obraditi ulaz veličine $\sqrt{10^8}$. Za 1 minut, algoritam A može obraditi ulaz veličine $\sqrt{10^8 \cdot 60} = \sqrt{6 \cdot 10^9} \approx 7.7 \cdot 10^4$

3. Traženo vreme izvršavanja algoritma dobija se rešavanjem jednačine:

$$n^3 = 10^6 \cdot t, \text{ za } n = 1000$$

$$1000^3 = 10^6 \cdot t$$

$$10^9 = 10^6 \cdot t$$

$$t = \frac{10^9}{10^6} = 10^3 \text{ sekundi}$$

Što je približno 16.6 minuta

4. Traženi broj instrukcija u sekundi dobija se rešavanjem jednačine:

$$n^7 = i \cdot 10, \text{ za } n = 100$$

$$100^7 = i \cdot 10$$

$$10^{14} = i \cdot 10^1$$

$$i = \frac{10^{14}}{10^1} = 10^{13} \text{ instrukcija u sekundi}$$

5. Kako je jednačina $T(n) = T(n-1) + T(n-2)$ homogena rekurentna jednačina drugog reda, njena karakteristiňa jednačina je:

$$t^n = t^{n-1} + t^{n-2}$$

$$t^n - t^{n-1} - t^{n-2} = 0$$

Pošto je jednačina drugog reda, pomnožićemo obe strane jednačine sa $\frac{t^2}{t^n}$

$$t^n - t^{n-1} - t^{n-2} = 0 / \cdot \frac{t^2}{t^n}$$

$$t^2 - t - 1 = 0$$

Opšte rešenje rekurentne jednačine je oblika $T(n) = c_1 \cdot t_1^n + c_2 \cdot t_2^n$ gde su t_1 i t_2 nule karakteristične jednačine a c_1 i c_2 realne konstante.

Pronadjimo nule jednačine $t^2 - t - 1 = 0$

$$t_{1,2} = \frac{-b \pm \sqrt{b^2 - 4 \cdot ac}}{2 \cdot a}$$

$$t_{1,2} = \frac{1 \pm \sqrt{1 - 4 \cdot (-1)}}{2} = \frac{1 \pm \sqrt{5}}{2}$$

$$t_1 = \frac{1 + \sqrt{5}}{2}, \ t_2 = \frac{1 - \sqrt{5}}{2}$$

Dakle, opšte rešenje polazne rekurentne jednačine je oblika:

$$T(n) = c_1 \cdot (\frac{1 + \sqrt{5}}{2})^n + c_2 \cdot (\frac{1 - \sqrt{5}}{2})^n$$

Ostaje da pronadjemo konstante c_1 i c_2. Kako su pošetni uslovi dati za $n = 0$ i $n = 1$, tj $T(0) = 0$ i $T(1) = 1$, napravićemo sistem gde ćemo n iz opšteg rešenja jednačine zameniti sa vrednostima za koje su nam početni uslovi poznati:

$$T(0) = c_1 \cdot (\frac{1+\sqrt{5}}{2})^0 + c_2 \cdot (\frac{1-\sqrt{5}}{2})^0 = 0$$
$$T(1) = c_1 \cdot (\frac{1+\sqrt{5}}{2})^1 + c_2 \cdot (\frac{1-\sqrt{5}}{2})^1 = 1$$

pa je sistem:

$$c_1 + c_2 = 0$$
$$c_1 \cdot \frac{1+\sqrt{5}}{2} + c_2 \cdot \frac{1-\sqrt{5}}{2}$$

Iz prve jednačine možemo zaključiti da je $c_1 = -c_2$, što kada se uvrsti u drugu jednačinu daje:

$$-c_2 \cdot \frac{1+\sqrt{5}}{2} - c_2 \cdot \frac{1-\sqrt{5}}{2} = 1$$

Ako obe strane jednačine pomnožimo brojem 2 dobijamo:

$$-c_2 \cdot \frac{1+\sqrt{5}}{2} - c_2 \cdot \frac{1-\sqrt{5}}{2} = 1 / \cdot 2$$
$$-c_2 \cdot (1+\sqrt{5}) - c_2 \cdot (1 = \sqrt{5}) = 2$$
$$c_2 \cdot (-(1+\sqrt{5}) + 1 - \sqrt{5}) = 2$$
$$c_2 \cdot (-1 - \sqrt{5} + 1 - \sqrt{5}) = 2$$
$$c_2 \cdot 2\sqrt{5} = 2$$
$$c_2 = -\frac{1}{\sqrt{5}}$$

Kada dobijenu vrednost c_2 uvrstimo nazad u prvu jednačinu dobijamo:

$$c_1 = \frac{1}{\sqrt{5}}$$

Konačno, opšte rešenje polazne rekurentne jednačine je:

$$T(n) = \frac{1}{\sqrt{5}} \cdot (\frac{1+\sqrt{5}}{2})^n - \frac{1}{\sqrt{5}} \cdot (\frac{1-\sqrt{5}}{2})^n = \frac{1}{\sqrt{5}} \cdot ((\frac{1+\sqrt{5}}{2})^n - (\frac{1-\sqrt{5}}{2})^n)$$

Što pripada $O((\frac{1+\sqrt{5}}{2})^n) \approx O(1.618^n)$

6. Jednačina $T(n) = T(n-1) + n + 1$ je nehomogena jednačina (zbog slobodnih članova n i 1) pa je potrebno svesti je na homogenu jednačinu narednim postupkom:

$$T(n) = T(n-1) + n + 1$$
$$T(n-1) = T(n-2) + (n-1) + 1$$

Oduzmimo drugu jednačinu od prve:

$$T(n) - T(n-1) = T(n-1) - T(n-2) + n - (n-1) + 2 - 2 = T(n-1) - T(n-2) + 1$$

$$T(n) = 2T(n-1) - T(n-2) + 1$$

Jednačina dobijena oduzimanjem je i dalje nehomogena jednačina, sada drugog reda, pa je potrebno ponoviti postupak:

$$T(n) = 2T(n-1) - T(n-2) + 1$$
$$T(n-1) = 2T(n-2) - T(n-3) + 1$$

Oduzimanjem druge jednačine od prve dobijamo:

$$T(n) - T(n-1) = 2T(n-1) - 2T(n-2) - T(n-2) + T(n-3) + 1 - 1$$
$$T(n) = 3T(n-1) - 3T(n-2) + T(n-3)$$

Dobijena jednačine je homogena jednačina trećeg reda pa je njena karakteristična jednačina:

$$t^n - 3t^{n-1} - 3t^{n-2} - t^{n-3} = 0$$

Kako je jednačina trećeg reda, karakterističnu jednačinu množimo sa $\frac{t^3}{t^n}$

$$t^n - 3t^{n-1} - 3t^{n-2} - t^{n-3} = 0 / \cdot \frac{t^3}{t^n}$$
$$t^3 - 3t^2 + 3t - 1 = 0$$

Nule dobijene kubne jednačine su $t_1 = t_2 = t_3 = 1$, tj dobijena nula je trostruka, pa je opšte rešenje rekurentne jednačine oblika:

$$T(n) = c_1 \cdot t_1^n + c_2 \cdot n \cdot t_2^n + c_3 \cdot n^2 \cdot t_3^n$$

tj

$$T(n) = c_1 \cdot t_1^n + c_2 \cdot n \cdot t_1^n + c_3 \cdot n^2 \cdot t_1^n$$
$$T(n) = c_1 \cdot 1^n + c_2 \cdot n \cdot 1^n + c_3 \cdot n^2 \cdot 1^n$$
$$T(n) = c_1 + c_2 \cdot n + c_3 \cdot n^2$$

Iz početnog uslova $T(1) = 1$ dobijamo:

$$T(1) = c_1 + c_2 + c_3 = 1$$

Kako jednačina ima tri nepoznate, potreban nam je sistem od tri jednačine. Dodatne dve jednačine dobijamo izračunavanjem vrednosti $T(2)$ i $T(3)$:

$$T(n) = T(n-1) + 2n + 1$$

$$T(2) = T(1) + 2 \cdot 2 + 1 = 1 + 4 + 1 = 6$$
$$T(3) = T(2) + 2 \cdot 3 + 1 = 6 + 6 + 1 = 13$$

Sada imamo sistem od tri jednačine sa tri nepoznate:

$$T(1) = c_1 + c_2 + c_3 = 1$$
$$T(2) = c_1 + 2c_2 + 4c_3 = 6$$
$$T(3) = c_1 = 3c_2 + 9c_3 = 13$$

Oduzimanjem prve jednačine od druge dobijamo:

$$c_1 - c_1 + 2c_2 - c_2 + 4c_3 - c_3 = 6 - 1$$

$$c_2 + 3c_3 = 5 \text{ (*)}$$

Oduzimanjem druge jednačine od treće dobijamo:

$$c_1 - c_1 + 3c_2 - 2c_2 + 9c_3 - 4c_3 = 13 - 6$$
$$c_2 + 5c_3 = 7 \text{ (**)}$$

Razlika (**) - (*) daje:

$$c_2 - c_2 + 5c_3 - 3c_3 = 7 - 5$$
$$2c_3 = 2$$
$$c_3 = 1$$

Uvrstimo dobijenu vrednost c_3 u (*):

$$c_2 + 3 \cdot 1 = 5$$
$$c_2 = 2$$

Dobijene vrednosti c_2 i c_3 uvrstimo u prvu jednačinu:

$$c_1 + 2 + 1 = 1$$
$$c_1 = -2$$

Konačno, opšte rešenje polazne rekurentne jednačine je:

$$T(n) = -2 + 2 \cdot n + 1 \cdot n^2$$

Što u O notaciji pripada $O(n^2)$

7. Prebrojimo sve naredbe koje se izvršavaju u okviru programskog koda:

```
k = 3;
for(i = 1; i < n; i++)
{
    while(k <= n)
    {
        k = k * 3;
        suma = suma + i;
    }
}
```

Dodela vrednosti promenljivoj k je jedan korak. Dodela u okviru inicijalizacije prve petlje se izvršava jednom, poredjenje $i < n$ se izvršava n puta, jer je brojaĭnicijalno postavljen na 1. Inkrementacija $i++$ se izvršava $n-1$ puta kao i naredbe unutar tela petlje.

Uslov druge petlje $k \leq n$ se izvršava $\lfloor \log_3 n \rfloor + 1$ puta a naredbe unutar tela petlje $\lfloor \log_3 n \rfloor$ puta. Naredbe unutar tela **while** petlje obuhvataju dve dodele, jedno množenje i jedno sabiranje.

Ukupan broj koraka je:

$$1 + 1 + n + (n-1) + (n-1) \cdot (\lfloor \log_3 n \rfloor + 1 + \lfloor \log_3 n \rfloor \cdot (1 + 1 + 1 + 1))$$
$$= 1 + 2n + (n-1) \cdot (5\lfloor \log_3 n \rfloor + 1) = 1 + 2n + 5n\lfloor \log_3 n \rfloor + n - 5\lfloor \log_3 n \rfloor - 1$$
$$= 3n + 5n\lfloor \log_3 n \rfloor - 5\lfloor \log_3 n \rfloor \text{ što pripada } O(n \log n)$$