Analiza efikasnosti algoritama I

- Asimptotske notacije
- Master metoda (teorema)

Asimptotske notacije (1/2)

- Služe za opis vremena izvršenja algoritma $T(n)$
  - gde je $n \in N$ veličina ulaznih podataka
  - npr. br. elemenata ulaznog niza
- Npr. $T(n) = an^2 + bn + c$
  - gde su a, b i c konstante
- Zanemarivanjem detalja ove funkcije može se proceniti da je asimptotsko vreme izvršenja tog algoritma neka funkcija od n^2
  - jer taj član najbrže raste sa n

Asimptotske notacije (2/2)

- Asimptotske notacije se mogu primeniti i na druge aspekte efikasnoti algoritma
  - Npr. zauzeće memorijskog prostora
- Ako se koristi za vreme, za koje vreme?
  - Vreme u najgorem slučaju (worst-case running time)
  - Vreme bez obzira na veličinu ulaznih podataka, itd.

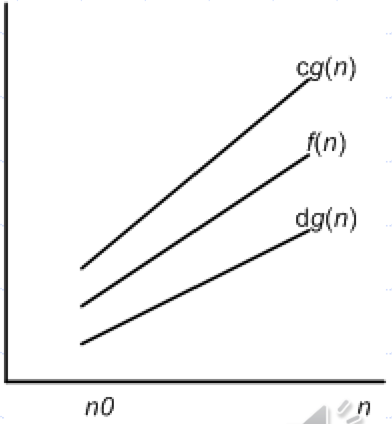
- Postoji pet asimptotskih notacije: Θ-notacija, O-notacija, Ω-notacija, o-notacija i ω-notacija

Θ-notacija (1/3)

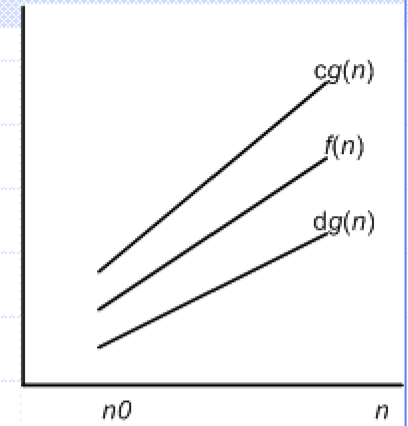
- Služi za određivanje vremena izvršenja algoritma u najgorem slučaju
- Def.: Za zadatu funkciju $g(n)$, $\Theta(g(n))$ je *skup funkcija*, $\Theta(g(n))=\{f(n)\}$, takvih da postoje pozitivne konstante c_1, c_2 i n_0 za koje je:

  $$0 \leq c_1 g(n) \leq f(n) \leq c_2 g(n), \text{ za svako } n \geq n_0$$

- Kažemo: $g(n)$ je ASIMPTOTSKI USKO OGRANIČENJE za $f(n)$
  - Umesto $f(n) \in \Theta(g(n))$ pišemo $f(n)=\Theta(g(n))$



Θ-notacija (2/3)



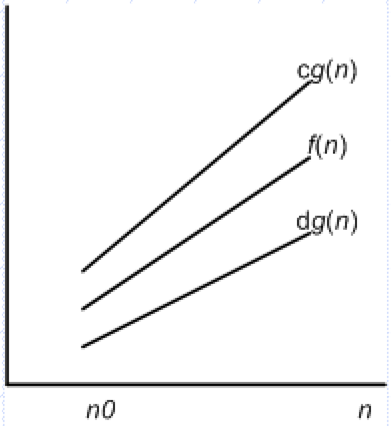
- $g(n)$ mora biti ASIMPTOTSKI NENEGATIVNO
- Θ-notacija se zasniva na odbacivanju članova nižeg reda i zanemarivanju koeficijenta uz vodeći član
  - Npr. da bi dokazali da je $n^2/2-3n=\Theta(n^2)$ potrebno je odrediti c_1, c_2 i n_0 tako da je za svako $n \geq n_0$:

$c_1 n^2 \leq n^2/2-3n \leq c_2 n^2 \Rightarrow$ rešenje $c_1=1/14$, $c_2=1/2$ i $n_0=7$

  - II primer: $6n^3 \neq \Theta(n^2)$. Dokaz kontradikcijom, pp da postoje c_2 i n_0 za koje je $6n^3 \leq c_2 n^2$, za svako $n \geq n_0$ $\Rightarrow n \leq c_2/6$, to je kontradikcija jer je c_2 konstanta

Θ-notacija (3/3)

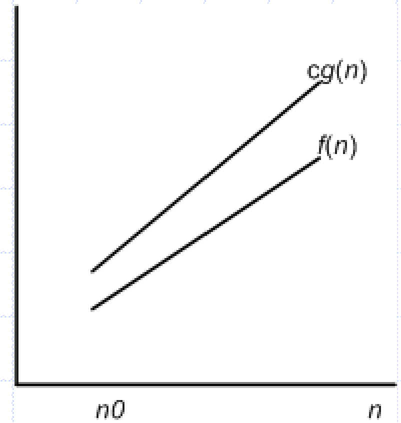
- Za $f(n)=an^2+bn+c$ je $f(n)=\Theta(n^2)$
  - Dokaz: iz def. Potrebno je izabrati konstante c_1=a/4, $c_2=7a/4$ i n_0=2max(|b|/a,(|c|/a)$^{1/2}$).
- Za bilo koji polinom $p(n)$ reda d, važi da je $p(n)=\Theta(n^d)$
  - Konstanta je polinom nultog reda: može se izraziti kao $\Theta(n^0)$ ili kao $\Theta(1)$



O-notacija (1/2)

- Služi za za definisanje ASIMPTOTSKI GORNJE GRANICE zadate funkcije
- Def.: za zadatu funkciju $g(n)$, $O(g(n))$ je *skup funkcija*, $O(g(n))=\{f(n)\}$, takvih da postoje pozitivne konstante c i n_0 za koje je:

  $0 \leq f(n) \leq cg(n)$, za svako $n \geq n_0$

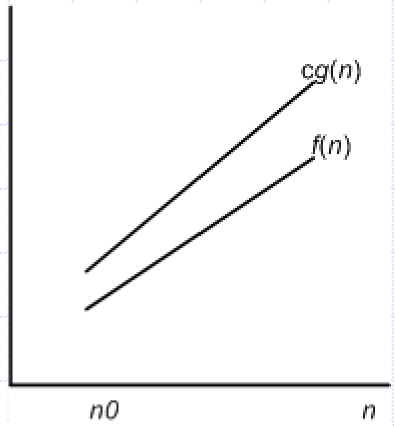


- $f(n)=an^2+bn+c$ je u $\Theta(n^2)$, takođe je i u $O(n^2)$
  - Ali i linearna funkcija $an+b$ je takođe u $O(n^2)$
  - Po teoriji skupova $\Theta(g(n)) \subseteq O(g(n))$

O-notacija (2/2)

- Može biti neobično da je npr. $n = O(n^2)$
- Vreme na osnovu ukupne strukture algoritma
  - Npr. za sledeći pseudo kod sa dve petlje očigledno je da je $O(n^2)$ gornja granica vremena izvršenja:

```
for(j=0; j<n; j++)
    for(k=0; (k<n) && condition; k++)
        // Neka O(1) obrada
```

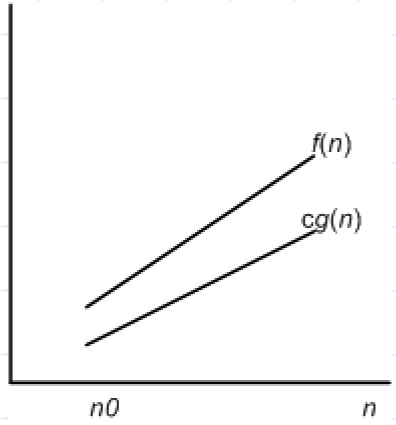


  - Ova granica ne implicira granicu $O(n^2)$ za svaki ulaz
    - Npr. ako je *condition* netačan, gornja granica je $O(n)$

Ω-notacija

- Služi za za definisanje ASIMPTOTSKI DONJE GRANICE zadate funkcije
- Def.: Za zadatu funkciju $g(n)$, $\Omega(g(n))$ je *skup funkcija*, $\Omega(g(n))=\{f(n)\}$, takvih da postoje pozitivne konstante c i n_0 za koje je:

  $0 \leq cg(n) \leq f(n)$, za svako $n \geq n_0$



- $\Omega(g(n))$ daje vreme izvršenja algoritma u najboljem slučaju (best-case running time)

Teorema o asimptotskim notacijama

- Za bilo koje dve funkcije $f(n)$ i $g(n)$, važi da je $f(n)=\Theta(g(n))$ ako i samo ako je $f(n)=O(g(n))$ i $f(n)=\Omega(g(n))$
  - Do asimptotski uske granice zadate funkcije na osnovu njene asimptotski gornje i donje granice
  - Obratno, na osnovu asimptotski uske granice odrediti asimptotski gornju i donju granicu

Asimptotske notacije u jednačinama i nejednačinama 1/3

- Već smo videli formulu $n = O(n^2)$
  - Slično bi mogli da napišemo $2n^2+3n+1=2n^2+\Theta(n)$
  - Kako se interpretiraju ovakve formule?

  - U $n = O(n^2)$, znak = označava: $n \in O(n^2)$
  - U većim formulama, asim. notacija predstavlja neku nepoznatu funkciju
  - Npr. formula $2n^2+3n+1=2n^2+\Theta(n)$ znači da je $2n^2+3n+1=2n^2+f(n)$, gde je $f(n)$ neka funkcija u skupu $\Theta(n)$

Asimptotske notacije u jednačinama i nejednačinama 2/3

- Broj anonimnih funkcija u nekom izrazu jednak broju pojava asimtotskih notacija.
  - Na primer, u izrazu $\Sigma\, O(i)$ postoji samo jedna anonimna funkcija, koja zavisi od i
- Šta ako se asimptotska notacija pojavljuje sa leve strane jednačine?
  - Npr. $2n^2 + \Theta(n) = \Theta(n^2)$
  - Pravilo: Kako god izabrali f-iju sa leve strane, postoji izbor f-ije sa desne strane, tako da je = zadovoljena
  - Za: $2n^2 + \Theta(n) = \Theta(n^2)$: za $f(n)$ postoji $g(n) \in \Theta(n^2)$, takva da je $2n^2 + f(n) = g(n)$ za svako n

Asimptotske notacije u jednačinama i nejednačinama 3/3

- Moguće je ulančati više ovakvih relacija
  - Npr. $2n^2+3n+1 = 2n^2 + \Theta(n) = \Theta(n^2)$

- Primenom gornjeg pravila svaka jednačina u lancu se interpretira nezavisno
  - Najpre: $2n^2+3n+1 = 2n^2 + f(n)$
  - A zatim: $2n^2 + g(n) = h(n)$, $g(n) \in \Theta(n)$, $h(n) \in \Theta(n^2)$
  - Zaključak: $2n^2+3n+1 = \Theta(n^2)$

o-notacija

- Granica koju daje O-notacija može, ali ne mora, biti asimptotski uska
  - Npr. $2n^2 = O(n^2)$ je asimptotski uska, $2n = O(n^2)$ nije
  - o-notacija služi za def. gornje granice koja **nije** uska
- Def.: $o(g(n))$ se definiše kao skup funkcija $o(g(n))=\{f(n)\}$, takvih da za svaku konstantu c postoji n_0 za koje je:

  $0 \leq f(n) < cg(n)$, za svako $n \geq n_0$

  - Npr. $2n = o(n^2)$, ali $2n^2 \neq o(n^2)$
  - Ako je $f(n) = o(g(n)) \Rightarrow$ limes $n \rightarrow \infty$ $f(n)/g(n) = 0$

ω-notacija

- ω-notacijom se označava donja granica koja nije asimptotski uska

- Def.: $\omega(g(n))$ se definiše kao skup funkcija $\omega(g(n))=\{f(n)\}$, takvih da za svaku konstantu c postoji n_0 za koje je:

  $0 \leq cg(n) < f(n)$, za svako $n \geq n_0$

  - Npr. $n^2/2 = \omega(n)$, ali $n^2/2 \neq \omega(n^2)$
  - Ako je $f(n)=\omega(g(n)) \Rightarrow$ limes $n \to \infty$ $f(n)/g(n) = \infty$

Poređenje funkcija (1/2)

- Relaciona svojstva Re brojeva važe i za asimptotska poređenja
  - Tranzitivnost za svih 5 asim. not.
  - Refleksivnost Θ, O, i Ω
  - Simetričnost za Θ, i transponovana sim O-Ω i o-ω
- Zato važi (a i b su Re brojevi):
  - $f(n)=\Theta(g(n))$ je kao $a=b$
  - $f(n)=O(g(n))$ je kao $a\leq b$
  - $f(n)=\Omega(g(n))$ je kao $a\geq b$
  - $f(n)=o(g(n))$ je kao $a<b$
  - $f(n)=\omega(g(n))$ je kao $a>b$

Poređenje funkcija (2/2)

- Kaže se da je:
  - $f(n)$ asimptotski manje od $g(n)$ ako je $f(n)=o(g(n))$ i
  - $f(n)$ asimptotski veće od $g(n)$ ako je $f(n)=\omega(g(n))$
- Trojakost ne važi!
  - Trojakost: za dva realna broja a i b, samo jedna od sledeće tri relacije može biti tačna: $a<b$, $a=b$ ili $a>b$
  - Ali, dve funkcije $f(n)$ i $g(n)$ mogu biti takve da za njih ne važi ni $f(n)=O(g(n))$ niti $f(n)=\Omega(g(n))$

Master metoda

- Recept za rešavanje rekurentne jednačine oblika
  $T(n) = aT(n/b) + f(n)$
  $a \geq 1$ i $b > 1$, $f(n)$ neka asimptotski pozitivna funkcija
- Master metoda razlikuje tri slučaja
  - Lako rešavanje mnogih rekurentnih jednačina
- Jednačina opisuje vreme izvršenja algoritma
  - koji deli problem veličine n na a podproblema
  - svaki veličine n/b se rešava rekurzivno
  - vreme rešavanja podproblema je $T(n/b)$
  - $f(n)$ pokriva cenu deljenja problema na podprobleme i kombinovanja rešenja tih podproblema

Master teorema

- Neka je data rekurentna jednačina oblika:

  $T(n) = aT(n/b) + f(n)$

  $a \geq 1$ i $b > 1$, $f(n)$ neka asimptotski pozitivna funkcija
- Tada $T(n)$ ima sledeće asimptotske granice:
  - Ako je $f(n) = O(n^{\log_b a-\varepsilon})$ za neku konstantu $\varepsilon > 0$, onda je $T(n) = \Theta(n^{\log_b a})$
  - Ako je $f(n) = \Theta(n^{\log_b a})$,
    onda je $T(n) = \Theta(n^{\log_b a} \cdot \lg n)$, lg je $\log_2$
  - Ako je $f(n) = \Omega(n^{\log_b a+\varepsilon})$ za neku konstantu $\varepsilon > 0$, i ako je $af(n/b) \leq cf(n)$ za neku konstantu $c < 1$ i za sva dovoljno velika n, onda je $T(n) = \Theta(f(n))$

Tumačenje master teoreme

- U sva tri slučaja $f(n)$ se poredi sa $n^{\log_b a}$
  - Veća od ove dve funkcije određuje rešenje
- Prilikom upoređivanja voditi računa o sledećem:
  - U slučaju 1, $f(n)$ mora biti polinomijalno manja, za faktor n^{ε}
  - U slučaju 3, $f(n)$ mora biti polinomijalno veća i mora zadovoljiti tzv. uslov regularnosti $af(n/b) \leq cf(n)$
- Tri slučaja ne pokrivaju sve mogućnosti!
  - Postoje procepi između slučaja 1 i 2, i slučaja 2 i 3

Korišćenje master metode (1/2)

- Prepozna se koji slučaj iz master teoreme važi, a onda se jednostavno napiše odgovor
- Primer 1: $T(n) = 9T(n/3) + n$
  - Rešenje: a=9, b=3, $f(n)=n$, $\log_b a=2$, $n^{\log_b a}=\Theta(n^2)$. Pošto je $n=O(n^{2-\varepsilon})$ za $\varepsilon=1$, u pitanju je slučaj 1, i rešenje je $T(n) = \Theta(n^2)$
- Primer 2: $T(n) = T(2n/3) + 1$
  - Rešenje: a=1, b=3/2, $f(n)=1$, $\log_b a=0$, $n^{\log_b a}=1$, Pošto je $1=\Theta(n^0)=\Theta(1)$, u pitanju je drugi slučaj i rešenje je $T(n) = \Theta(\lg n)$

Korišćenje master metode (2/2)

- Primer 3: $T(n) = 3T(n/4) + n \lg n$
  - Rešenje: a=3, b=4, $f(n)= n \lg n$, $\log_b a$=0.793, $n^{\log_b a}=\Theta(n^{0.793})$. Pošto je $n \lg n=\Omega(n^{0.793+\varepsilon})$ za $\varepsilon\approx 0.2$, u pitanju je slučaj 3, ako je zadovoljen uslov regularnosti. Za dovoljno veliko n je: $af(n/b)=3(n/4)\lg(n/4)\leq(3/4)n \lg n = cf(n)$ za c=3/4. Pošto je uslov regularnost ispunjen, rešenje je $T(n) = \Theta(n \lg n)$.