Függvények nagyságrendje

Adatstruktúrák és algoritmusok

1. gyakorlat

Definíció

Ha $f, g:\ \mathbb{N} \to \mathbb{R}^+$ függvények, akkor

i. $f = O(g)$ jelöli azt a tényt, hogy léteznek olyan $c, n_0 > 0$ állandók, hogy minden $n \geq n_0$ esetén $f(n) \leq c \cdot g(n)$ teljesül.

ii. $f = \Omega(g)$ jelöli azt a tényt, hogy léteznek olyan $c, n_0 > 0$ állandók, hogy minden $n \geq n_0$ esetén $f(n) \geq c \cdot g(n)$ teljesül.

iii. $f = \Theta(g)$ jelöli azt a tényt, hogy $f = O(g)$ és $f = \Omega(g)$ is teljesül.

1. Bizonyítsuk be, hogy

   (a) $n^2 - 4n + 7 = \Theta(n^2)$;

   (b) $1 + 2 + \ldots + n = \Theta(n^2)$;

   (c) $\log_{10} n = \Theta(\log_2 n)$;

   (d) $2^n = O(3^n)$, de $2^n \neq \Omega(3^n)$;

   (e) $2^{n+1} = \Theta(2^n)$, de $2^{2n} \neq O(2^n)$;

   (f) $1 + 2 + 2^2 + \ldots + 2^n = \Theta(2^n)$;

   (g) $100(n-1)! = O(n!)$, de $100(n-1)! \neq \Omega(n!)$.

2. Adjunk minél jobb O becslést a következő függvényekre.

   (a) $(n^2 + 8)(n + 1)$

   (b) $5^n + n^n + n!$

   (c) $\dfrac{n^4 + n^2 + 1}{n^4 + 1}$

   (d) $\dfrac{n^3 + 5\log(n^{4n})}{n^4 + 1}$

   (e) $2n\sqrt{n} + 3n^2 + 10n\log_2^3 n$

3. Van egy számítógépes programunk, ami egy 50 méretű feladaton a jelenlegi gépünkön 1 nap alatt fut le. Beszereztünk egy százszor gyorsabb számítógépet. Ugyanazon programmal mekkora feladatot lehet az új gépen egy nap alatt megoldani, ha a program lépésszáma n méretű feladat esetén

   (a) n, (b) n^3, (c) 2^n?

4. Mi lehet $T(n)$, ha $T(1) = 2$ és $n \geq 2$ esetén $T(n) = 3 \cdot T(n-1) + 1$?

5. Egy problémára két algoritmusunk van.

   Az $\mathcal{A}$ algoritmus az $n \geq 2$ méretű problémából 10 lépéssel 2 darab $n-1$ méretűt készít és ezeket oldja meg rekurzívan.

   A $\mathcal{B}$ algoritmus az $n \geq 2$ méretű problémából 3 lépéssel 4 darab $n-1$ méretűt készít és ezeket oldja meg rekurzívan.

   Az $n = 1$ esetben mindkét eljárás 1 lépést használ.

   Melyik algoritmus lesz nagy n értékekre gyorsabb?