فهرست مطالب

فصل ۱

معرفی نمادهای مجانبی

۱ نمادهای مجانبی

ابزارهائی که توسط آنها می توان زمان اجرا و یا حافظه گرفته شده دو یا چند الگوریتم را با هم مقایسه نماییم.

۱.۱ نماد $big - O$

اگر $f : N \to R^+$ باشد آنگاه:

$$O(f(n)) = \{g : N \to R^+ | \exists c \in R^+, \forall n \in N(g(n) \leq cf(n)\}$$

$$f(n) = \begin{cases} n^۳ & n < ۱۰۰۰ \\ ۲n^۲ & n \geq ۱۰۰ \end{cases}$$

$$O(n^۲) = \{n^۲, n\, lg\, n, f(n), \ldots\}$$

در شکل قبل حتی اگر f_2 صد برابر شود باز هم از $O(f_1)$ است. در شکل بالا اگر f_1 در یک ضریب ضرب شود داریم $f_2 \in O(f_1)$، پس چیزی که مهم است ضریب c نیست، مهم درجه است.

$big-O$ همان مفهوم کران بالا را دارد مثلاً برای حافظه مصرفی، $big-O$ کران بالای مصرف حافظه است. پس الگوریتمی مفید است که کران بالایی زمان اجرای آن پایین باشد.

۲.۱ نماد $big-\Omega$

متن

۱.۲.۱ زیر زیر بخش

۲ نماد θ

۳ نماد $small-o$

۱.۳ زیر بخش

۱.۱.۳ زیر زیر بخش

۴ نماد $small-\omega$

اگر $f : N \to R^+$ آنگاه:

$$\omega(f(n)) = \{g : N \to R^+ | \forall c > 0 \quad n \in N \quad g(n) \geq cf(n)\}$$

مثال: نشان دهید:

$$\omega(f)n)) \cap o(f(n)) = \emptyset$$

حل: اگر $g(n) \in \omega(f(n)) \cap o(f(n))$ آنگاه برای $d_\circ > \circ$ یک $M_۱$ یک $M_۱$ چنان موجود است که:

$$\forall n M_۱ \quad g(n) \geq d_\circ f(n) \quad (g(n) \in \omega(f(n)))$$

حال برای $c_\circ = \frac{d_\circ}{۲}$ یک M چنان موجود است که:

$$\forall n \geq M_۲ \quad g(n) \leq c_\circ f(n) \quad (g(n) \in o(f(n))) \Rightarrow g(n) \leq \frac{d_\circ}{۲} f(n) \Rightarrow$$
$$\forall n \geq MAX\{M_۱, M_۲\} \quad \frac{d_\circ}{۲} f(n) \geq d_\circ f(n) \Rightarrow d_\circ \geq ۲d_\circ \Rightarrow d_\circ \leq \circ$$

پس به تناقض رسیدیم، بنابراین حکم ثابت می شود.

۵ قضیه ماکزیمم گیری

۱.۵ زیر بخش