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:

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

$$f(n) = \begin{cases} n^3 & n < 1000 \\ 2n^2 & n \geq 100 \end{cases}$$

$$O(n^2) = \{n^2, n\, lg\, n, f(n), \ldots\}$$

در شکل قبل حتی اگر $f_۲$ صد برابر شود باز هم از $O(f_۱)$ است. در شکل بالا اگر $f_۱$ در یک ضریب ضرب شود داریم $f_۲ \in O(f_۱)$، 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

۲ نماد θ

۳ نماد $small-o$

۴ نماد $small-\omega$

اگر $f : N \to R^+$ آنگاه:

$$\omega(f(n)) = \{g : N \to R^+ | \forall c > ۰ \quad n \in N \quad g(n) \geq cf(n)\}$$

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 $d_۰ > ۰$ یک $M_۱$ یک $M_۱$ چنان موجود است که:

$$\forall n M_۱ \quad g(n) \geq d_۰ f(n) \quad (g(n) \in \omega(f(n)))$$

حال برای $c_۰ = \frac{d_۰}{۲}$ یک M چنان موجود است که:

$$\forall n \geq M_۲ \quad g(n) \leq c_۰ f(n) \quad (g(n) \in o(f(n))) \Rightarrow g(n) \leq \frac{d_۰}{۲} f(n) \Rightarrow$$
$$\forall n \geq MAX\{M_۱, M_۲\} \quad \frac{d_۰}{۲} f(n) \geq d_۰ f(n) \Rightarrow d_۰ \geq ۲d_۰ \Rightarrow d_۰ \leq ۰$$

پس به تناقض رسیدیم، بنابراین حکم ثابت می شود.

۵ قضیه ماکزیمم گیری