

فصل ۱

معرفی نمادهای مجانبی

۱ نمادهای مجانبی

ابزارهایی 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 $f : N \rightarrow R^+$ باشد آنگاه:

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

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

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

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

۲ نماد θ

۳ نماد $small - o$

۴ نماد $small - \omega$

اگر $f : N \rightarrow R^+$ آنگاه:

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

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 $d_0 > 0$ یک M_1 یک M_2 چنان موجود است که:

$$\forall n M_1 \quad g(n) \geq d_0 f(n) \quad (g(n) \in \omega(f(n)))$$

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

$$\forall n \geq M_{\gamma} \quad g(n) \leq c_* f(n) \quad (g(n) \in o(f(n))) \Rightarrow g(n) \leq \frac{d_*}{\gamma} f(n) \Rightarrow$$

$$\forall n \geq MAX\{M_1, M_{\gamma}\} \quad \frac{d_*}{\gamma} f(n) \geq d_* f(n) \Rightarrow d_* \geq \gamma d_* \Rightarrow d_* \leq 0.$$

پس به تناقض رسیدیم، بنابراین حکم ثابت می شود.

۵ قضیه ماکزیمم گیری