"مقدمه"

کارایی یک الگوریتم به مقدار زمان، حافظه و سایر منابع مورد نیاز برای اجرا بستگی دارد. کارایی یک الگوریتم با بررسی پیچیدگی و مرتبه زمانی و حافظه‌ای و کمک نمادهای مجانبی اندازه‌گیری می‌شود. یک الگوریتم ممکن است عملکرد یکسانی برای انواع مختلف ورودی نداشته باشد و با افزایش اندازه ورودی، عملکردش تغییر کند. مطالعه تغییر در عملکرد الگوریتم با تغییر در ترتیب اندازه ورودی به عنوان تحلیل مجانبی تعریف می‌شود. در این مقاله با بررسی این مفاهیم و تعریفشان یاد خواهیم گرفت که چگونه کارایی یک الگوریتم را از نظر مرتبه زمانی توصیف کنیم.

"نمادهای مجانبی"

نمادهای مجانبی، نمادهای ریاضی هستند که برای توصیف زمان اجرای یک الگوریتم در زمانی که ورودی به سمت یک مقدار خاص یا یک مقدار محدود میل می‌کند استفاده می‌شود؛ به عنوان مثال در مرتب سازی حبابی، زمانی که آرایه ورودی از قبل مرتب شده است، زمان صرف شده توسط الگوریتم خطی است،

یعنی بهترین حالت؛ اما زمانی که آرایه ورودی در شرایط معکوس قرار دارد الگوریتم حداکثر زمان را برای مرتب سازی عناصر یعنی بدترین حالت صرف می‌کند. وقتی آرایه ورودی نه مرتب شده است و نه به ترتیب معکوس، آن‌گاه به زمان متوسطی نیاز دارد. این مدت زمان‌ها با استفاده از نمادهای مجانبی نشان داده می‌شوند.

- عمدتاً سه نماد مجانبی وجود دارد:

- نماد Big-O
- نماد امگا (Ω)
- نماد تتا (Θ)

در ادامه هر کدام از این نمادها را تعریف و بررسی می‌کنیم.

نماد Big-O یا (O)

در نظریه پیچیدگی محاسباتی نماد O بزرگ (Big O notation) برای نشان دادن رابطه میان تعداد داده ها و منابع محاسباتی مورد نیاز برای حل یک مسئله با استفاده از یک الگوریتم استفاده می شود. استفاده از این نماد معمولاً برای بررسی زمان و یا حافظه مورد نیاز برای حل مسئله ای با تعداد زیادی ورودی می باشد.

در ریاضیات علامت O بزرگ رفتار حدی یک تابع را وقتی آرگومان های آن به یک عدد خاص یا به بی نهایت میل می کنند، توصیف می کند. علامت O بزرگ به کاربر اجازه می دهد که تابع را ساده کند تا بر روی نرخ رشد آن متمرکز شود. بنابراین توابع مختلف با نرخ رشد یکسان می توانند دارای یک علامت O مشابه باشند.

بدترین حالت یا حالت میانگین زمان اجرا یا حافظه مورد استفاده یک الگوریتم اغلب به صورت تابعی از طول داده ورودی با استفاده از علامت O بزرگ توصیف می شود. این به طراحان الگوریتم اجازه می دهد که رفتار الگوریتم هایشان را پیش بینی کنند و تصمیم بگیرند که کدام الگوریتم را استفاده کنند.

برخلاف نماد Θ که یک تابع را از بالا و پایین به صورت مجانبی (حدی) محدود می کند

نماد O یک کران بالای حدی مشخص می کند. وقتی تابعی را با استفاده از علامت

O بزرگ توصیف می کنیم به طور معمول تنها یک کران بالا برای نرخ رشد آن تابع

فراهم می کنیم.

علامت O بزرگ در دو دامنه کاربرد دارد:

• در ریاضیات معمولاً برای نشان دادن این که یک سری هندسی متناهی تا چه

اندازه به تابع مورد نظر نزدیک است، خصوصاً در مورد سری تیلور ناقص از این

علامت استفاده می شود.

• در علوم کامپیوتر این علامت در تحلیل الگوریتم ها کاربرد دارد.

در هر دو کاربرد تابع $g(x)$ که در $O(\dots)$ به گونه ای انتخاب می شود که تا حد

امکان ساده باشد

$$\begin{cases} T(n) \in O(g(n)) \Leftrightarrow \exists\, n_0, c > 0 : \forall n \geq n_0 ,\ T(n) \leq c \cdot g(n) \\ \\ T(n) \in O(g(n)) \Leftrightarrow \lim\limits_{n \to \infty} \dfrac{T(n)}{g(n)} = \begin{cases} c & \text{عدد ثابت} \\ 0 \end{cases} \end{cases}$$

نماد امگا یا (Ω)

نماد امگا نشان دهنده مرز پایینی زمان اجرای یک الگوریتم است بنابراین بهترین حالت پیچیدگی یک الگوریتم را مشخص می کند. این نماد به صورت زیر تعریف می شود:

$\Omega(g(n)) = f(n)$ که ثابت های مثبت c و n_0 وجود دارد به طوری که $0 \le cg(n) \le f(n)$ برای همه $n \geq n_0$. عبارت فوق را می توان به عنوان تابع $f(n)$ از مرتبه $\Omega(g(n))$ توصیف کرد، اگر یک ثابت مثبت c وجود داشته باشد به طوری که برای n به اندازه ی کافی بزرگ، $f(n)$ بالای $cg(n)$ باشد. برای هر مقدار n، حداقل زمان مورد نیاز الگوریتم توسط امگا $\Omega(g(n))$ داده می شود.

$$\begin{cases} T(n) \in \Omega(g(n)) \Leftrightarrow \exists n_0, c > 0 : \forall n \geq n_0, T(n) \geq c.g(n) \\ \\ T(n) \in \Omega(g(n)) \Leftrightarrow \lim\limits_{n \to \infty} \dfrac{T(n)}{g(n)} = \begin{cases} c \\ \infty \end{cases} \end{cases}$$

نماد تتا یا (Θ)

نماد تتا تابع را از بالا و پایین محدود می‌کند. از آن جایی که این نماد کران بالایی و پایینی زمان اجرای یک الگوریتم را نشان می‌دهد، برای تجزیه و تحلیل پیچیدگی میانگین حالت یک الگوریتم استفاده می‌شود. برای تابع $\Theta(g(n))$، $g(n)$ با رابطه زیر به دست می‌آید:

$f(n) = \Theta(g(n))$ ثابت های مثبت n_0، c_1 و c_2 وجود دارد به طوری که

$c_1 gn \leq f(n) \leq c_2 gn \leq 0$

برای همه $n \geq n_0$.

یعنی اگر ثابت های مثبت c_1 و c_2 وجود داشته باشد، به طوری که برای n های به اندازه کافی بتوان $f(n)$ را بین $c_1 g(n)$ و $c_2(g(n))$ قرار داد، می‌توان آن را به عنوان تابع $f(n)$ از مرتبه $\Theta(g(n))$ توصیف کرد. اگر تابع $f(n)$ در جایی بین $c_1 g(n)$ و $c_2 g(n)$ برای همه $n \geq n_0$ قرار گیرد آنگاه به $f(n)$ مجانبی کران دار گفته می‌شود.

$$\begin{cases} T(n) \in \Theta(g(n)) \Leftrightarrow \exists n_0, c_1, c_2 > 0 : \forall n \geq n_0, c_1 \cdot g(n) \leq T(n) \leq c_2 \cdot g(n) \\ T(n) \in \Theta(g(n)) \Leftrightarrow \lim\limits_{n \to \infty} \frac{T(n)}{g(n)} = c \end{cases}$$