ساختمان داده‌ها و الگوریتم‌ها

نیم‌سال اول ۰۰ـ۹۹
مدرس: مسعود صدیقین

دانشکده‌ی مهندسی کامپیوتر

یادآوری جلسه ششم — تحلیل روابط بازگشتی - روش تقسیم و حل

در جلسه قبل، راجع به تحلیل روابط بازگشتی با استفاده از قضیه اصلی صحبت کردیم. فرم کلی قضیه اصلی به این صورت است که اگر داشته باشیم:

$$T(n) = aT(n/b) + f(n),$$

در این صورت، سه حالت زیر را داریم:

۱. اگر $f(n) = O(n^{\log_b^a - \epsilon})$ آنگاه $T(n) = \theta(n^{\log_b^a})$.

۲. اگر $f(n) = \theta(n^{\log_b^a})$ آنگاه $T(n) = \theta(n^{log_b^a} \log n)$.

۳. اگر $f(n) = \Omega(n^{\log_b^a + \epsilon})$ آنگاه $T(n) = \theta(f(n))$.

در واقع، این قضیه فرم کلی نتیجه‌ای بود که با استفاده از درخت بازگشت به دست آوردیم. به عنوان نمونه، برای تابع $T(n) = 5T(n/2) + n^2$ داریم:

$$T(n) = \theta(n^{\log_2^5}).$$

همچنین در این جلسه، در رابطه با روش‌های تقسیم و غلبه صحبت کردیم. این روش‌ها، از دو بخش تقسیم و غلبه تشکیل شده است. در یک الگوریتم تقسیم و غلبه، در بخش تقسیم مساله اصلی به تعدادی زیر مساله تقسیم شده و هر کدام از آنها به صورت بازگشتی با استفاده از همان الگوریتم حل می‌شوند. سپس در بخش غلبه، مساله اصلی با استفاده از زیرمساله‌های حل شده حل می‌شوند. به عنوان نمونه، روش ضرب کاراتسوبا را مورد بررسی قرار دادیم.

مساله: فرض کنید دو عدد n رقمی ذخیره شده در دو آرایه با اندازه n (یک عدد در هر خانه از آرایه) داریم و می‌خواهیم این دو عدد را در هم ضرب کنیم.

ضرب معمولی با استفاده از روش دستی را می‌توان در زمان $O(n^2)$ محاسبه کرد. با این حال، به دنبال روش بهتری هستیم. با استفاده از روش تقسیم غلبه، می‌توان این مساله را به این روش حل کرد:

- آرایه A و B را هر کدام به دو زیر آرایه A_h, A_l و B_h, B_l تقسیم کنید. از آنجا که $A = A_l + A_h * 10^{n/2}$ و $B = B_l + B_h * 10^{n/2}$ است، داریم:

$$A * B = A_hB_h * 10^n + A_lB_l + (A_hB_l + A_lB_h) * 10^{n/2} \qquad (1)$$

- به طور بازگشتی A_hB_h و A_lB_l و همینطور $(A_h + A_l)(B_h + B_l)$ را محاسبه می‌کنیم.
- مقدار $A_hB_l + A_lB_h$ را از روی مقدارهای محاسبه شده در قسمت دوم محاسبه می‌کنیم.
- مقدار $A * B$ را با استفاده از فرمول ۱ محاسبه می‌کنیم.

زمان اجرای این الگوریتم هم شامل حل ۳ زیرمساله بازگشتی با اندازه $n/2$ است به همراه تعدادی عملیات به صورت $O(n)$ که شامل ضرب در توانی از ۱۰ و همینطور جمع اعدادی با طول حداکثر $2n$ است. فلذا:

$$T(n) = 3T(n/2) + O(n) = O(n^{\log_2^3})$$

پرسش: ممکن است اندازه $A_h + A_l$ و $B_h + B_l$ برابر با $n/2 + 1$ شود. آیا می‌توانید برای حل این مشکل، عملیاتی از $O(n)$ انجام دهید که یک رقم از هر دو کم شده و بخش بازگشتی دقیقا برابر با $T(n/2)$ شود؟ لازم به تحویل دادن پاسخ نیست. به این سوال در خلوت خودتان فکر کنید!