

Алгоритмы и структуры данных-1

SET 2. Задача A4.

Осень 2024. Клычков М. Д.

Пункт 1. Для начала стоит вспомнить, что нам известно об асимптотике алгоритма Штрассена. На лекции было доказано, что $T_{\text{Ш}} = O(n^{\log_2 7})$. От нас требуется найти алгоритм, выражающийся рекуррентным соотношением

$$T(n) = a \cdot T\left(\frac{n}{4}\right) + \Theta(n^2) \quad (1)$$

Заметим, что для данного рекуррентного соотношения возможно применение мастер-теоремы.

$$\text{при } \log_4 a = 2 \Leftrightarrow a = 16: T(n) = O(n^2 \log_4 n) \quad (2)$$

$$\text{при } \log_4 a > 2 \Leftrightarrow a > 16: T(n) = O(n^{\log_4 a}) \quad (3)$$

$$\text{при } \log_4 a < 2 \Leftrightarrow a < 16: T(n) = O(n^2) \quad (4)$$

На лекции отмечалось, что умножение квадратных матриц может быть реализовано только за $O(n^{2+\varepsilon})$, $\varepsilon > 0$. Воспользуемся этой информацией и уберем из рассмотрения (??), так как в этом случае $T(n) = O(n^2)$.

Покажем, что нам подходит случай (??). Достаточно, чтобы $T(n) = O(T_{\text{Ш}}(n))$ или, подставив конкретную $T(n)$, $O(n^2 \log_4 n) = O(n^{\log_2 7})$

$$\begin{aligned} n^2 \log_4 n &\leq cn^{\log_2 7} \\ \log_4 n &\leq cn^{\log_2 7 - 2} \\ \log_4 n &\leq cn^{\log_2 1.75} \Leftrightarrow \log n = O(n^{\log_2 1.75}) \end{aligned}$$

А последнее мы принимаем за истину (знания о порядке роста функций).

Рассмотрим оставшийся случай (??). Тут все действия будут схожи с предыдущим пунктом, только остается решить уравнение с параметром. Нам нужно найти такие a , что $T(n) = O(T_{\text{Ш}}(n))$ или, подставив конкретную $T(n)$, $O(n^{\log_4 a}) = O(n^{\log_2 7}) \Leftrightarrow \log_4 a \leq \log_2 7$

$$\begin{aligned} \frac{1}{2} \log_2 a &\leq \log_2 7 \\ \log_2 a &\leq \log_2 49 \\ a &\leq 49 \end{aligned}$$