МФТИ, ФПМИ

Алгоритмы и структуры данных, осень 2021

Семинар №1. Асимптотики, простые алгоритмы

Мастер-теорема. Пусть $T(n) = aT(n/b) + f(n)$, где $a \geqslant 1, b > 1$. Тогда:

1. если $\exists \varepsilon > 0 : f(n) = O(n^{\log_b a - \varepsilon})$, то $T(n) = \Theta(n^{\log_b a})$;
2. если $f(n) = \Theta(n^{\log_b a})$, то $T(n) = \Theta(n^{\log_b a} \cdot \log n)$;
3. если $\exists \varepsilon > 0 : f(n) = \Omega(n^{\log_b a + \varepsilon})$, причём $\exists c < 1 : af(n/b) \leqslant cf(n)$, то $T(n) = \Theta(f(n))$.

1. Пусть $c > 0$ — константа. Докажите, что решение рекурренты $T(n) = T(n/3) + T(2n/3) + cn$ ведёт себя как $\Omega(n \log n)$.

2. Найдите решение для каждого из приведённых ниже рекуррентных соотношений в терминах Θ:

а) $T(n) = 2T(n/3) + 1$;

б) $T(n) = 5T(n/4) + n$;

в) $T(n) = 7T(n/7) + n$;

г) $T(n) = 9T(n/3) + n^2$;

д) $T(n) = 8T(n/2) + n^3$;

е) $T(n) = 49T(n/25) + n^{3/2} \log n$;

ж) $T(n) = T(n-1) + 2$;

з) $T(n) = T(n-1) + n^c$, где $c \geqslant 1$ — константа;

и) $T(n) = T(n-1) + c^n$, где $c > 1$ — константа;

к) $T(n) = 2T(n-1) + 1$;

л) $T(n) = T(n/2) + 1$;

м) $T(n) = T(\sqrt{n}) + 1$.

3. Истина или ложь?

а) $2^{n+3} = \Theta(2^n)$;

б) $2^n = \Theta(2^{n/2})$;

в) $n^2 = O(2^n)$;

г) $\frac{n}{\log n} = \Omega(\log n)$;

д) $\frac{n}{\log n} = \Theta(\frac{n}{2})$;

е) $\sqrt{n}^{\sqrt{n}} = O((\log n)^n)$;

ж) $n^{\log n} = O(1.1^n)$;

з) $n^n = O(n!)$.

4. Пусть f, g, s, t — функции из $\mathbb{N}$ в $\mathbb{N}$. Пусть известно, что $f(n) = \xi(s(n))$, а $g(n) = \eta(t(n))$, где ξ и η — какие-то из значков O, Ω, Θ. Как можно связать $f(n) \cdot g(n)$ и $s(n) \cdot t(n)$?

5. Пусть $a_1, \ldots, a_n$ и $b_1, \ldots, b_m$ — две последовательности чисел. Предложите алгоритм их слияния (относительный порядок элементов в обеих последовательностях должен сохраниться) для получения лексикографически минимальной/максимальной последовательности за $O(n+m)$. Считаем, что список чисел $x_1, \ldots, x_k$ лексикографически меньше списка $y_1, \ldots, y_k$, если существует такое $m < k$, что $x_1 = y_1, \ldots, x_m = y_m$, но $x_{m+1} < y_{m+1}$. Можете считать, что все данные числа попарно различны.

6. Пусть $a = (a_1, \ldots, a_n)$, $b = (b_1, \ldots, b_m)$ — две последовательности. Говорят, что a является подпоследовательностью b, если из b можно вычеркнуть некоторые элементы так, чтобы получилась a (без изменения порядка оставшихся элементов). Формальнее, a является подпоследовательностью b, если существует набор $1 \leqslant i_1 < i_2 < \ldots < i_n \leqslant m$, такой что $b_{i_j} = a_j$ для всех $j \in \{1, \ldots, n\}$. За $O(m)$ определите, является ли a подпоследовательностью b.

7. По данному числу n найдите все пары целых положительных чисел (a, b), такие что $a \leqslant b \leqslant n$, и $a \mid b$. Оцените асимптотическое поведение числа таких пар.

8. Число 0 записано в n-разрядной двоичной системе. К нему $2^n - 1$ раз прибавляется единица. Будем считать, что время, необходимое на прибавление единицы, равно количеству единиц в двоичной записи числа, которые становятся нулями. Оцените среднюю сложность всех таких операций. Какие операции являются самыми дешёвыми, а какие — самыми дорогими?

9. Изначально есть массив $a_1, a_2, \ldots, a_n$. К нему применяются q преобразований вида l, r, x, что означает, что числа с l-го по r-е нужно увеличить на x. Выведите массив после всех преобразований. Асимптотика: $O(n+q)$.

10. На столе лежит n куч бананов, в i-й из них — a_i бананов. Обезьянка может выбрать произвольное число S и есть по S бананов в минуту. Однако съедаемые за минуту бананы должны изначально лежать в одной куче. Если в куче бананов меньше S, то за минуту съедаются они все. Хозяин зоопарка вернётся через M минут, а обезьянка хочет растянуть удовольствие и есть как можно дольше. Найдите минимальное S такое, что обезьянка успеет съесть все бананы до прихода хозяина. Асимптотика: $O(n \log(\max\limits_{i=1}^{n} a_i))$.

1. Докажем, что $T(n) \geqslant \alpha n \ln n$ индукцией по n. Для n имеем: $T(n) \geqslant \frac{\alpha}{3} n \ln(n/3) + \frac{2\alpha}{3} n \ln(2n/3) + cn$, хотим продолжить $\geqslant \alpha n \ln n$. Раскроем логарифмы, останется неравенство на α, выполнение которого завершает доказательство.

2. В большинстве пунктов можно напрямую воспользоваться мастер-теоремой. Для убедительности можно доказать, что получаемая ей оценка действительно является верной (так мы можем "угадать" ответ и доказать, что он подходит). В пункте м): если $n = 2^k$, то T делим пополам степень двойки каждый раз при спуске в рекурсию. Делить пополам можно $\log k$ раз, то есть $T(n) = \Theta(\log \log n)$.

3.

а) да;
б) нет;
в) да;
г) да;
д) нет;
е) да;
ж) да;
з) нет.

4. Если $f = O(s)$ и $g = \Omega(t)$, то однозначно ничего утверждать нельзя (приведите примеры!). Иначе из значков выбирается самый слабый: $O \cdot O = O, O \cdot \Theta = O, \Theta \cdot \Theta = \Theta, \Theta \cdot \Omega = \Omega, \Omega \cdot \Omega = \Omega$.

5. Первым нужно выписать наименьшее из a_1 и b_1. Какое — следующим?

6. Сработает жадный алгоритм.

7. Без доказательства можно использовать факт, что $\sum\limits_{k=1}^{n} \frac{1}{k} = \Theta(\log n)$.

8. Для каждого бита посчитайте количество раз, когда он изменит своё значение с единицы на ноль.

9. Перейдите к массиву соседних разностей. Как он изменяется после каждого преобразования? Как по нему восстановить исходный массив?

10. Вспомните бинарный поиск.