Рябых Владислав, Б05-905

Алгоритмы и модели вычислений.
Домашнее задание № 1

Задача 2. Имеются различные клетчатые таблички — нужно подсчитать число способов замостить ими большое поле из клеток без пробелов и наложений.
Разрешено использовать таблички: чёрный квадрат 2×2, белый квадрат 2×2, серый прямоугольник 2×1 с возможностью поворота. Поле представляет собой полосу $2 \times n$. найдите асимптотику числа замощений и явную формулу для него.

Решение. Пусть $T(n)$ — количество способов замостить полосу $2 \times n$. Для нахождения рекурсивной формулы воспользуемся следующим наблюдением: так как размеры наших табличек не превышают 2×2, то принципиально у нас есть 2 способа замостить край полосы:

1. замостить один крайний столбец: на это у нас есть всего один способ — поставить серый прямоугольник 2×1

2. замостить сразу два крайних столбца: на это у нас есть 3 способа — чёрный квадрат 2×2, белый квадрат 2×2, два серых прямоугольника 1×2 каждый (случай двух серых прямоугольников 2×1 покрывается предыдущим пунктом, поэтому тут не рассматривается)

Таким образом, получаем $T(n) = T(n-1) + 3T(n-2)$
Найдём асимптотику $T(n)$ исходя из того, что $T(n-1) = T(n-2) + 3T(n-3) \geqslant T(n-2)$:

- $T(n) = T(n-1) + 3T(n-2) \leqslant 4T(n-1) = 4 \cdot 4T(n-2) = \ldots = 4^n \Rightarrow T(n) = O(4^n)$
- $T(n) = T(n-1) + 3T(n-2) \geqslant 4T(n-2) = 4 \cdot 4T(n-2\cdot 2) = \ldots = 4^{\frac{n}{2}} = 2^n \Rightarrow T(n) = \Omega(2^n)$

Но ведь это ужасно грубые оценки, а дальше Вы найдёте точную

Решим линейное рекуррентное соотношение: для начала запишем характеристическое уравнение

$$\lambda^2 = \lambda + 3 \quad \Rightarrow \quad \lambda_{1,2} = \frac{1 \pm \sqrt{13}}{2}$$

Таким образом $T(n) = C_1 \left(\dfrac{1+\sqrt{13}}{2}\right)^n + C_2 \left(\dfrac{1-\sqrt{13}}{2}\right)^n$, подберём константы из начальных условий: при $n = 0$ у нас есть 1 способ замостить ленту, при $n = 1$ — также 1 способ, то есть $T(0) = T(1) = 1$, тогда

$$C_1 + C_2 = 1 \qquad C_1 \cdot \frac{1+\sqrt{13}}{2} + C_2 \cdot \frac{1-\sqrt{13}}{2} = 1$$

Решая, получаем $C_1 = \dfrac{1}{2} + \dfrac{1}{2\sqrt{13}} = \dfrac{1}{\sqrt{13}}\lambda_1$ и $C_2 = \dfrac{1}{2} - \dfrac{1}{2\sqrt{13}} = -\dfrac{1}{\sqrt{13}}\lambda_2$
И, окончательно:

$$T(n) = \frac{1}{\sqrt{13}} \cdot \left(\frac{1+\sqrt{13}}{2}\right)^{n+1} - \frac{1}{\sqrt{13}} \cdot \left(\frac{1-\sqrt{13}}{2}\right)^{n+1} \Rightarrow T(n) = \Theta\left(\left(\frac{1+\sqrt{13}}{2}\right)^n\right)$$

+0,9

Задача 3. Найдите Θ асимптотику рекуррентности, которая определяется в следующем тексте.
Colour the edges of a complete graph of n vertices by three colours so that the number of triangles all whose edges get a different colour is maximal. Denote this maximum by $G_3(k)$. They conjectured that $G_3(k)$ is obtained as follows: clearly $G_3(1) = G_3(2) = 0, G_3(3) = 1, G_3(4) = 4$. Suppose $G_3(k_1)$ has already been determined for every $k_1 < k$. Then

$$G_3(k) = G_3(u_1) + G_3(u_2) + G_3(u_3) + G_3(u_4) + u_1u_2u_3 + u_1u_2u_4 + u_1u_3u_4 + u_2u_3u_4$$

where $u_1 + u_2 + u_3 + u_4 = k$ and the u's are as nearly equal as possible.

Решение. «u's are as nearly equal as possible», так что $u_{min} \geqslant \frac{k}{5}, u_{max} \leqslant \frac{k}{3}$, поэтому

- $G_3(k) \geqslant G_{min}(k) = 4G_3\left(\frac{k}{5}\right) + \frac{4}{5^3}k^3$
- $G_3(k) \leqslant G_{max}(k) = 4G_3\left(\frac{k}{3}\right) + \frac{4}{3^3}k^3$

Решим:

- $G_{min}(k) = 4G_3\left(\frac{k}{5}\right) + \frac{4}{5^3}k^3 = 4\cdot\left(4G_3\left(\frac{k}{25}\right) + \frac{4}{5^3}\left(\frac{k}{5}\right)^3\right) + \frac{4}{5^3}k^3 = \ldots = \left(1 + \frac{4}{5^3} + \ldots + \left(\frac{4}{5^3}\right)^{\log k}\right)\cdot \frac{4}{5^3}k = \Omega(k^3)$
  Таким образом, $G_3(k) = \Omega(k^3)$
- Аналогично $G_{max}(k) = 4G_3\left(\frac{k}{3}\right) + \frac{4}{3^3}k^3 = 4\cdot\left(4G_3\left(\frac{k}{9}\right) + \frac{4}{3^3}\left(\frac{k}{3}\right)^3\right) + \frac{4}{3^3}k^3 = \ldots =$
  $= \left(1 + \frac{4}{3^3} + \ldots + \left(\frac{4}{3^3}\right)^{\log k}\right)\cdot \frac{4}{3^3}k = O(k^3)$
  Таким образом, $G_3(k) = O(k^3)$

И, окончательно, $G_3(k) = \Theta(k^3)$ +1

Задача 4. (i) Вычислите число правильно составленных скобочных выражений, содержащих n скобок, в которых в любом непустом *собственном* префиксе число открывающих скобок больше числа закрывающих.
(ii) Найдите явное аналитическое выражение для производящей функции чисел BR_{4n+2} правильных скобочных последовательностей длины $4n + 2$ (ответ в виде суммы ряда не принимается).

Решение. Пусть $CBS(n)$ — язык всех правильных скобочных последовательностей (Correct bracket sequences) из n пар скобок, $L(n) \subseteq CBS(n)$ — язык из CBS, в которых в любом непустом префиксе число открывающих скобок больше числа закрывающих, $w \in \Sigma^*, \Sigma = \{), (\}$ — произвольное слово
Докажем, что $w \in L(n) \Leftrightarrow w = (u), u \in CBS(n-1), n \geqslant 2$:

- $\Leftarrow$: так как $u \in CBS(n-1)$, то $\forall x \in pref(u) \hookrightarrow |x|_{(} \geqslant |x|_{)}$, а так как каждый префикс w представляется в виде $y = (x$, то $\forall y \in pref(w) \hookrightarrow |y|_{(} > |y|_{)} \Rightarrow w \in L(n)$

- $\Rightarrow$: очевидно, $w[1] = ($, $w[2n] =)$ в силу того, что $w \in CBS(n)$. Таким образом, $w = (u)$. Осталось показать, что u — правильная скобочная последовательность. Если бы это на самом деле было не так, то существовал бы $x \in pref(u) : |x|_{(} < |x|_{)}$, но тогда существовал бы $y = (x \in pref(w) : |y|_{(} = |x|_{(} + 1 \leqslant |x|_{)} = |y|_{)}$ — противоречие и таким образом $u \in CBS(n-1)$, что и требовалось

Таким образом мы доказали, что $|L(n)| = |CBS(n-1)| = C_{(n-1}$, где C_n — n-й элемент последовательности чисел Каталана +0,5

***Задача* 5.** Оцените трудоемкость рекурсивного алгоритма, разбивающего исходную задачу размера n на три задачи размером $\lceil \frac{n}{\sqrt{3}} \rceil - 5$, используя для этого $10\frac{n^3}{\log n}$ операций.

Решение. $T(n) = 3T\left(\left\lceil \frac{n}{\sqrt{3}} \right\rceil - 5\right) + 10\frac{n^3}{\log n}$

- $T(n) \geqslant 10\frac{n^3}{\log n} \Rightarrow T(n) = \Omega\left(\frac{n^3}{\log n}\right)$

- $T(n) \leqslant 3T\left(\left\lceil \frac{n}{\sqrt{3}} \right\rceil\right) + 10\frac{n^3}{\log n}$; применим мастер теорему: $d = \log_b a = \log_{\sqrt{3}} 3 = 2$; $10\frac{n^3}{\log n} = \Omega(n^{d+\varepsilon})$ при ε, равном, например, 0.5 $\Rightarrow$ по мастер-теореме $3T\left(\left\lceil \frac{n}{\sqrt{3}} \right\rceil\right) + 10\frac{n^3}{\log n} = \Theta\left(\frac{n^3}{\log n}\right) \Rightarrow T(n) = O\left(\frac{n^3}{\log n}\right)$

Таким образом, получаем, $T(n) = \Theta\left(\frac{n^3}{\log n}\right)$ +1

***Задача* 6.** Рассмотрим детерминированный алгоритм поиска медианы по кальке известного линейного алгоритма, где используется разбиение массива на четвёрки элементов, в каждой из которых определяется *нижняя* медиана, т. е. из в каждой четверки выбирается второй по порядку элемент (элементы можно считать различными). Приведите рекуррентную оценку числа сравнений в этой процедуре и оцените сложность такой модификации.

Решение. Пусть x — опорный элемент, тогда в половине четвёрок элементов есть по 2 элемента, больших x, за исключением, быть может, последней группы, в которой менее 4-х элементов. Таким образом, элементов, больших x, по крайней мере $3 \cdot \left(\frac{1}{2} \cdot \frac{n}{4} - 2\right) = \frac{3n}{8} - 6$; а элементов, меньших x, соответственно $2 \cdot \left(\frac{1}{2} \cdot \frac{n}{4} - 2\right) = \frac{n}{4} - 4$

Таким образом, $T(n) = T\left(\left\lceil \frac{n}{4} \right\rceil\right) + T\left(\frac{\cancel{5}^{3}n}{\cancel{8}_{4}} + \cancel{6}^{4}\right) + O(n)$ а асимптотика?

***Задача* 7.** Функция натурального аргумента $S(n)$ задана рекурсией:

$$S(n) = \begin{cases} 100 & , n \leqslant 100 \\ S(n-1) + S(n-3) & , n > 100 \end{cases}$$

Оцените число рекурсивных вызовов процедуры $S(\cdot)$ при вычислении $S(10^{12})$

А это как вообще получено?

Решение. Рассматриваем функцию $S(n)$ при больших n: $S(n) = S(n-1) + S(n-3) \Rightarrow S(n-1) = S(n-3) + S(n-5) \geqslant S(n-3)$. Таким образом при вычислении $S(n-1)$ происходит больше рекурсивных вызовов, чем при вычислении $S(n-3)$. Пусть $K(m)$ — количество рекурсивных вызовов процедуры $S(\cdot)$ при вычислении $S(m)$. Тогда по условию нам надо оценить $K(10^{12})$, а только что мы показали, что $K(n-1) \geqslant K(n-3)$, следовательно $2K(n-3) \leqslant K(n) \leqslant 2K(n-1)$ Таким образом получаем:

Это снова какая-то супергрубая оценка, как в задаче 2. На самом деле число вызовов удовлетворяет почти такой же рекурренте $T(n)=T(n-1)+T(n-3)+1 \Rightarrow$ её надо порешать

$$2^{\frac{10^{12}-100}{3}} \leqslant \ldots \leqslant 2K(10^{12}-3) \leqslant K(10^{12}) \leqslant 2K(10^{12}-1) \leqslant \ldots \leqslant 2^{10^{12}-100}$$

Задача 9. Оцените трудоемкость рекурсивного алгоритма, разбивающего исходную задачу размера n на n задач размеров $\lceil \frac{n}{2} \rceil$ каждая, используя для этого $O(n)$ операций.

Решение. $T(n) = nT\left(\left\lceil \dfrac{n}{2} \right\rceil\right) + O(n)$. $d = \log_b a = \log_2 n$; $O(n) = O(n^{d-\varepsilon})$ при ε, равном, например, $0.5 \Rightarrow$ по мастер-теореме $T(n) = \Theta(n^d) = \Theta(n^{\log n})$

Ответьте себе на вопрос, почему она там не работает