

Алгоритмы и модели вычислений.

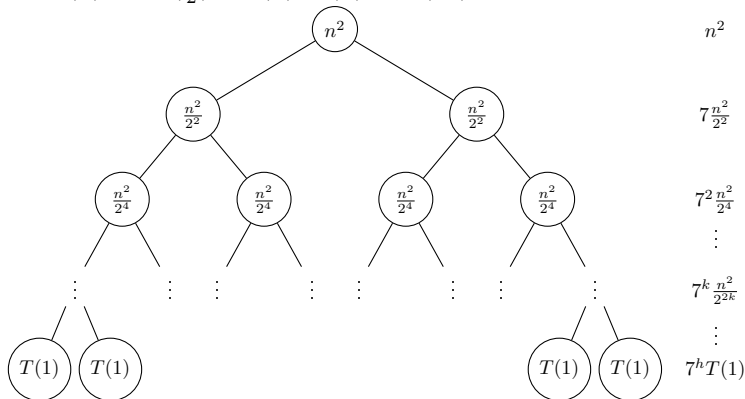
Задание 2: Арифметические операции и линейные рекуррентные последовательности

Сергей Володин, 272 гр.

задано 2014.02.20

(каноническое) Задача 6

$T(n) = 7T(\frac{n}{2}) + f(n)$, $f(n) = O(n^2)$. Дерево рекурсии:



Высота дерева $h = \log_2 n$. $T(n) = \sum_{k=0}^{h-1} 7^k f(\frac{n}{2^k}) + 7^h T(1) \leq$. Из определения $O \exists C > 0 \exists n_0: \forall n \geq n_0 \hookrightarrow f(n) \leq Cn^2$, откуда

первая сумма $\sum_{k=0}^{h-1} 7^k f(\frac{n^2}{2^{2k}}) \leq Cn^2 \sum_{k=0}^{h-1} (\frac{7}{4})^k = Cn^2 \frac{(7/4)^h - 1}{7/4 - 1} = C_1 n^2 ((7/4)^{\log_2 n} - C_2) = C_1 n^2 n^{\log_2 \frac{7}{4}} - C_3 n^2 = C_1 n^2 n^{\log_2 7} - C_3 n^2 = C_1 n^{\log_2 7} - C_3 n^2$.

Второе слагаемое $7^h T(1) = 7^{\log_2 n} T(1) = Cn^{\log_2 7}$

Поэтому $T(n) \leq n^{\log_2 7} - C_5 n^2$ Ответ: $T(n) = O(n^{\log_2 7})$