# Алгоритмы и модели вычислений.
# Задание 1: Алгоритмы и оценка сложности

Сергей Володин, 272 гр.

задано 2014.02.13

## (каноническое) Задача 1

$f(n) \overset{\text{def}}{=} \sum\limits_{i=1}^{n} \frac{1}{i}$, $g(n) = \log n$. Доказать: $f = \Theta(g) \Leftrightarrow \begin{cases} f = O(g) \\ g = O(f) \end{cases} \Leftrightarrow \begin{cases} \exists C_1, n_1 \colon \forall n \geqslant n_1 \hookrightarrow f(n) \leqslant C_1 g(n) & (1) \\ \exists C_2, n_2 \colon \forall n \geqslant n_2 \hookrightarrow g(n) \leqslant C_2 f(n) & (2) \end{cases}$

1. Докажем утверждение: пусть $f(n), g(n) \colon \exists n_0, C_1 > 0 \colon \forall n \geqslant n_0 \hookrightarrow \underbrace{f(n+1) - f(n)}_{\Delta_f(n)} \leqslant C_1 \underbrace{g(n+1) - g(n)}_{\Delta_g(n)}$. Тогда $f = O(g)$. Действительно, выберем $C_2 > 0$ таким образом, что $f(n_0) \leqslant C_2 g(n_0)$ (всегда можно сделать). Возьмем $C$ для определения $O$ как $C = \max(C_1, C_2)$. Докажем по индукции $\forall n \geqslant n_0 \hookrightarrow f(n) \leqslant Cg(n)$:
   (a) $f(n_0) \leqslant C_2 g(n_0) \leqslant C g(n_0) \blacksquare$
   (b) Пусть $f(n) \leqslant Cg(n)$. Докажем для $n+1$: по условию $\Delta_f(n) = f(n+1) - f(n) \leqslant C_1(g(n+1) - g(n)) \leqslant C(g(n+1) - g(n))$. Перегруппируем, получим $f(n+1) - Cg(n+1) \leqslant f(n) - Cg(n) \overset{\text{предп.}}{\leqslant} 0$, т.е. $f(n+1) \leqslant Cg(n+1) \blacksquare$
2. Докажем (1).
   (a) $\measuredangle \Delta_f(n) \overset{\text{def}}{=} f(n+1) - f(n) = \frac{1}{n+1} \leqslant \frac{1}{n}$.
   (b) $\measuredangle \Delta_g(n) \overset{\text{def}}{=} g(n+1) - g(n) = \log(n+1) - \log n = \log \frac{n+1}{n} = \log(1 + \frac{1}{n}) = \frac{1}{n} + \bar{\bar{o}}(\frac{1}{n}) = \boxed{*}$, $n \to \infty$. Но по определению $\bar{\bar{o}}$ $\exists n_1 \colon \forall n \geqslant n_1 \hookrightarrow \boxed{*} \geqslant \frac{1}{n}(1 - \frac{1}{2}) = \frac{1}{2}\frac{1}{n}$. Тогда $\frac{1}{n} \leqslant 2\boxed{*} = 2(g(n+1) - g(n))$
   (c) Получаем $\Delta_f(n) = f(n+1) - f(n) \overset{2a}{\leqslant} \frac{1}{n} \overset{2b}{\leqslant} 2(g(n+1) - g(n)) = 2\Delta_g(n)$, и по 1 получаем $f = O(g)$.
3. Докажем (2).
   (a) $\measuredangle \Delta_f(n) = \frac{1}{n+1}$. Докажем, что это больше, чем $\frac{1}{2}\frac{1}{n}$: $\frac{1}{n+1} - \frac{1}{2}\frac{1}{n} = \frac{2n-n-1}{2n(n+1)} = \frac{n-1}{2n(n+1)} \geqslant 0$, $n \geqslant 1$. Итак, $\Delta_f(n) \geqslant \frac{1}{2}\frac{1}{n}$
   (b) $2b \Rightarrow \Delta_g(n) = \frac{1}{n} + \bar{\bar{o}}(\frac{1}{n}) \leqslant \frac{1}{n}(1 + \frac{1}{2})$ при $n \geqslant n_2 > 1$. Значит, $\frac{3}{2}\frac{1}{n} \geqslant \Delta_g(n)$
   (c) $\Delta_g(n) \overset{3b}{\leqslant} \frac{3}{2}\frac{1}{n} \overset{3a}{\leqslant} \frac{3}{2} \cdot 2 \cdot \Delta_f(n)$ при $n \geqslant n_2$, и по 1 получаем $g = O(f)$.

## (каноническое) Задача 2

## (каноническое) Задача 3

1. $T(n) = 9T(\frac{n}{3}) + f(n)$, $f(n) = \Theta(n^2 \log n)$.
   (a) Докажем, что теорема неприменима. $a = 9, b = 3 \Rightarrow \log_b a = \log_3 9 = 2$.
      i. Если $\exists \varepsilon > 0 \colon f(n) = O(n^{2-\varepsilon})$, то $\exists C > 0 \exists n_0$, для $n \geqslant n_0$ получим $f(n)/n^{2-\varepsilon} \leqslant C > 0$, то есть $n^2 \log n / n^{2-\varepsilon} \equiv n^{\varepsilon} \log n \leqslant C$, что неверно (функция неограничена сверху).
      ii. Если $f = \Theta(n^2)$, то $\exists n_0 \exists C > 0 \colon f \leqslant Cn^2$ для $n \geqslant n_0$, и $\log n \leqslant C$, что неверно (функция неограничена сверху).
      iii. Если $\exists \varepsilon > 0 \colon f = \Omega(n^{2-\varepsilon})$, то $\exists n_0 \colon \forall n \geqslant n_0 \hookrightarrow f \geqslant Cn^{2+\varepsilon}$, и $\log n \geqslant Cn^{\varepsilon}$, откуда $\frac{\log^n}{n^{\varepsilon}} \geqslant C > 0$, что неверно, так как $\forall \varepsilon > 0 \hookrightarrow \lim\limits_{n \to \infty} \frac{\log n}{n^{\varepsilon}} = +0$
   (b) Найдем ответ через дерево рекурсии. В корне ($i = 0$) выполняется $n^2 \log n$ операций, у каждой вершины 9 детей, на уровне $i+1$ $n_{i+1} = n_i/3$. У листьев (по индукции по высоте дерева) $1 = n_h = \frac{n}{3^h}$, поэтому высота дерева *(не считая корня)* $h = \log_3 n$. Найдем суммарное время:
   $$T(n) = \Theta(n^2 \log n + 9(\frac{n}{3})^2 \log \frac{n}{3} + 9^2(\frac{n}{3^2})^2 \log \frac{n}{3^2} + ... + 9^{h-1}(\frac{n}{3^{h-1}})^2 \log \frac{n}{3^{h-1}}) + 9^h T(1)$$
   Найдем сумму в аргументе $\Theta$: $\sum\limits_{i=0}^{h-1} 9^i (\frac{n}{3^i})^2 \log \frac{n}{3^i} = n^2 \sum\limits_{i=0}^{h-1} (\log n - i \log 3) = n^2 \log n (h-1) - n^2 \frac{h-1}{2} \log 3 =$
   $= n^2 \log n(\log_3 n - 1) - n^2 \frac{\log_3 n - 1}{2} \log 3 = n^2 \log^2 n - n^2 \log n - n^2 \log n + Cn^2 = \Theta(n^2 \log^2 n)$.
   Найдем $9^h T(1) = C 9^{\log_3 n} = Cn^2$. Имеем $T(n) = \Theta(n^2 \log^2 n) + Cn^2 = \boxed{\Theta(n^2 \log^2 n)}$
2. $T(n) = 16T(\frac{n}{4}) + f(n)$, $f(n) = \Theta(n^2)$. $a = 16$, $b = 4$. Применим второй пункт Теоремы: $\Theta(n^{\log_b a}) \equiv \Theta(n^2)$, поэтому $f(n) = \Theta(n^{\log_b a})$, и отсюда $T(n) = \boxed{\Theta(n^2 \log n)}$.

3. $T(n) = 4T(\frac{n}{2}) + \Theta(\frac{n^2\sqrt{n}}{\log^2 n})$. $a = 4$, $b = 2 \Rightarrow \log_b a = 2$. Возьмем $\varepsilon = \frac{1}{4}$ и применим третий пункт Теоремы: $f(n) \stackrel{?}{=} \Omega(n^{2+\varepsilon})$. Рассмотрим $\frac{f(n)}{n^{2+\varepsilon}} = \frac{n^2\sqrt{n}}{n^2 n^\varepsilon \log^2 n} = \frac{n^{\frac{1}{2}-\varepsilon}}{\log^2 n} = \frac{n^{1/4}}{\log^2 n} = (\frac{n^{1/8}}{\log_n})^2 \xrightarrow{n\to\infty} +\infty$, поэтому $\exists C > 0 \exists n_0 > 0 \colon \forall n \geqslant n_0 \hookrightarrow f(n) \geqslant Cn^{2+\varepsilon}$.

   Значит, оценка верна, и по теореме получаем $T(n) = \boxed{\Theta(\frac{n^{5/2}}{\log^2 n})}$

Сравним первую и вторую функции: $\frac{n^2 \log^2 n}{n^2 \log n} = \log n \xrightarrow{n\to\infty} +\infty$, поэтому первый алгоритм хуже. Сравним вторую и третью функции: $\frac{n^2\sqrt{n}}{\log^2 n}\frac{1}{n^2 \log n} = \frac{n^{1/2}}{\log^3 n} = (\frac{n^{1/6}}{\log n})^3 \xrightarrow{n\to\infty} +\infty$, поэтому третий алгоритм хуже.
Ответ: $\boxed{\text{второй алгоритм}}$ имеет наименьшую асимптотическую стоимость.

## (каноническое) Задача 4

## (каноническое) Задача 5

## Задача 1

1. $T(n) = 2T(\frac{n}{3}) + \Theta(n^2)$

## Задача 2

## Задача 3