Задание 1. Оценки асимптотики. Рекурсивные алгоритмы.

1 Алгоритм получает на вход число n (в десятичной записи) и создаёт массив $A[2,\ldots,n]$, заполненный нулями. Далее алгоритм выполняет следующую процедуру, пока массив не окажется заполнен единицами. Идёт по массиву от 2 до n пока не встретит первый ноль. Пусть ноль оказался в ячейке с номером k. Тогда алгоритм выводит k и заполняет все ячейки с номерами, кратными k, единицами: идёт по массиву дальше с шагом один и через каждые k клеток записывает в ячейку единицу.

Какую последовательность чисел выводит алгоритм?

Оцените временную сложность алгоритма.

Является ли алгоритм полиномиальным?

2 Верно ли, что

$$\textbf{a)}\ n = O(n\log n);\ \ \textbf{b)}\ \exists\varepsilon > 0 : n\log n = \Omega(n^{1+\varepsilon})?$$

3 Известно, что $f(n) = O(n^2),\ g(n) = \Omega(1),\ g(n) = O(n)$. Положим

$$h(n) = \frac{f(n)}{g(n)}.$$

1. Возможно ли, что **а)** $h(n) = \Theta(n\log n)$; **б)** $h(n) = \Theta(n^3)$?

2. Приведите наилучшие (из возможных) верхние и нижние оценки на функцию $h(n)$ и приведите пример функций $f(n)$ и $g(n)$ для которых ваши оценки на $h(n)$ достигаются.

4 Найдите Θ-асимптотику $\sum\limits_{i=1}^{n}\sqrt{i^3 + 2i + 5}$.

5 Пусть для положительной функции $f(n)$ известно, что $f(n) = (3+o(1))^n + \Theta(n^{100})$. Верно ли в общем случае, что $\log f(n) = \Theta(n)$?

6* Оцените ассимптотику роста функции $f(n) = \binom{2n}{n}$.

7 Вычислите $7^{13} \mod 167$, используя алгоритм быстрого возведения в степень.

8 Функции $T_1(n)$ и $T_2(n)$ заданы рекуррентными формулами, известно что $T_i(1) = T_i(2) = T_i(3) = 1, i = 1, 2$.

1. Найдите асимтотику роста функции $T_1(n) = T_1(n-1) + cn$ (при $n > 3$);

2. Докажите, что для функции $T_2(n) = T_2(n-1) + 4T_2(n-3)$ (при $n > 3$) справедлива оценка $\log T_2(n) = \Theta(n)$.

3*. Найдите (точную) асимтотику роста функции $T_2(n)$.

9* Предложите $O(\sqrt{m}\log m)$ алгоритм нахождения длины периода десятичной дроби $\frac{n}{m}$. Докажите его корректность и оцените асимптотику.

10* $f(1) = g(1) = 1$ $f(n) = a\cdot g(n-1) + b\cdot f(n-1)$ $g(n) = c\cdot g(n-1) + d\cdot f(n-1)$ где a, b, c, d положительные константы. Предложите алгоритм вычисляющий $f(n)$ со сложностью $O(\log n)$ арифметических операций.