

Домашнее задание

1. Алгоритм⁴ получает на вход число n (в десятичной записи) и создаёт массив $A[2, \dots, n]$, 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.

1. Какую последовательность чисел выводит алгоритм?
2. Оцените временную сложность алгоритма.
3. Является ли алгоритм полиномиальным?

2. Докажите, что для произвольной константы $c > 0$ функция $g(n) = 1 + c + c^2 + \dots + c^n$ есть

- (a) $\Theta(1)$, если $c < 1$;
- (b) $\Theta(n)$, если $c = 1$;
- (c) $\Theta(c^n)$, если $c > 1$.

3. Верно ли, что **a)** $n = O(n \log n)$? **б)** $\exists \varepsilon > 0 : n \log n = \Omega(n^{1+\varepsilon})$?

4. 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 **a)** $h(n) = \Theta(n \log n)$; **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.
5. Дана программа

```
for (bound = 1; bound < n; bound *= 2) {
    for (i = 0; i < bound; i += 1) {
        for (j = 0; j < n; j += 2)
            печать ("алгоритм")
        for (j = 1; j < n; j *= 2)
            печать ("алгоритм")
    }
}
```

Пусть $g(n)$ обозначает число слов “алгоритм”, которые напечатает соответствующая программа. Найдите Θ -асимптотику $g(n)$.

⁴Алгоритм в этой задаче можно оптимизировать, но это не должно вас смущать. В нашем курсе (как и стандартно в математических курсах) нужно решать сформулированную задачу, поэтому, пожалуйста, удержитесь от порывов оптимизировать алгоритм и оценивать его сложность, если этого не требуется в условии.

Во всех задачах ниже мы полагаем, что арифметические операции стоят $O(1)$.

6 [Шень 1.3.1 (а,б,г)]. Постройте линейный по времени онлайн-алгоритм, который вычисляет следующие функции или укажите индуктивные расширения для следующих функций:

- а) среднее арифметическое последовательности чисел;
- б) число элементов последовательности целых чисел, равных её максимальному элементу;
- в) максимальное число идущих подряд одинаковых элементов;

Комментарий: В книге А. Шеня «Программирование. Теоремы и задачи» приведены задачи с решениями. Если в задаче ДЗ указана ссылка на эту книгу, то не нужно техать решение этих задач. Их нужно решить самостоятельно и свериться с решением в книге.

7. Дано три отсортированных по возрастанию массива, внутри каждого массива все элементы различные. Предложите⁵ линейный алгоритм нахождения числа различных элементов в объединении массивов.

8 [Шень 1.3.4]. Дана последовательность целых чисел a_1, a_2, \dots, a_n . Необходимо найти её самую длинную строго возрастающую подпоследовательность. Предложите **а)** $O(n^2)$ алгоритм (докажите его корректность и асимптотику); **б)** $O(n \log n)$ алгоритм.

9*. На вход подаётся последовательность натуральных чисел x_1, \dots, x_n в которой один из элементов встречается строго больше, чем $\frac{n}{2}$ раз. Постройте алгоритм, который находит этот элемент, и при этом может использовать в качестве внешней памяти только стек (в который можно помещать только элементы последовательности), операции со стеком стоят $O(1)$ времени; в оперативной памяти программа использует $O(1)$ битов памяти и $O(1)$ регистров (в каждом из которых может храниться число x_i).

Числа x_i идут потоком данных на вход и каждое доступно для считывания только один раз — вернуться обратиться к прочитанным ранее числам можно, только если сохранить их в памяти.

⁵Здесь и всюду далее мы требуем не только описание алгоритма, но и доказательство его корректности, а также доказательство оценок на время работы алгоритма.