

Практика по алгоритмам

Александр Мишунин, Алексей Давыдов*

Осень, 2015

*Составители сборника не являются авторами самих задач. Авторы не указаны в учебных целях.

1 Практика 1. Асимптотика

1.1 Практика

Все функции: $\mathbb{N} \rightarrow \mathbb{N}$

- $f = O(g) \Leftrightarrow \exists_{N,C>0} \forall_{n \geq N} f(n) \leq C \cdot g(n)$
- $f(n) = \Omega(g(n)) \Leftrightarrow g(n) = O(f(n))$
- $f(n) = \Theta(g(n)) \Leftrightarrow f(n) = O(g(n)) \wedge f(n) = \Omega(g(n))$
- $f(n) = o(g(n)) \Leftrightarrow \forall_{C>0} \exists_N \forall_{n \geq N} f(n) < C \cdot g(n)$
- $f(n) = \omega(g(n)) \Leftrightarrow g(n) = o(f(n))$

1. (a) $\max(f(n), g(n)) = \Theta(f(n) + g(n))$?
(b) $\min(f(n), g(n)) = \Theta(f(n) + g(n))$?
(c) $f(n) = O(f(n)^2)$?
(d) $f(n) = \Omega(f(n/2))$?
(e) $f(n) = O(g(n)) \Rightarrow \log f(n) = O(\log g(n))$?
(f) $f(n) = O(g(n)) \Rightarrow 2^{f(n)} = O(2^{g(n)})$?
(g) $f(n) = o(g(n)) \Rightarrow \log f(n) = o(\log g(n))$?
(h) $f(n) = o(g(n)) \Rightarrow 2^{f(n)} = o(2^{g(n)})$?
(i) $\sum_{k=1}^n \frac{1}{k} = \Omega(\log n)$?
(j) $\sum_{k=0}^{\log n} \lceil n/2^k \rceil = O(n^2)$?
(k) $\prod_{k=1}^n (2 \cdot 4^k) = \Theta(1)$?

2. Во всех пунктах нужно ответить на вопрос “за сколько работает”:

(a)

- `for (a = 1; a < n; a++)`
- `for (b = 0; b < n; b += 1)`
- ;

(b)

- `for (a = 1; a < n; a++)`
- `for (b = 0; b < n; b += a)`
- ;

(c) Найти такие a, b, c : $abc = n$, $a + b + c = \min$. Решение:

- `for (a = 1; a <= n; ++a)`
- `for (b = 1; a * b <= n; ++b)`
- `c = N / a / b, ... ;`

(d) Еще одно решение задачи (c):

- `for (a = 1; a * a * a <= n; ++a)`
- `for (b = 1; b * b <= n; ++b)`
- `c = N / a / b, ... ;`

(e) И еще одно решение задачи (c):

- `for (a = 1; a * a * a <= n; ++a)`
- `for (b = a; a * b * b <= n; ++b)`
- `c = N / a / b, ... ;`

(f) Дополнительный вопрос: что делает этот код?

- `a = 1, b = n;`
- `while (a < b) {`
- `while (x[a] < M && a <= b) a++;`
- `while (x[b] > M && a <= b) b--;`
- `if (a <= b) swap(x[a++], x[b--]);`
- `}`

(g) Дополнительный вопрос: а если бы вместо 2 было бы 1?

- `while (a >= 2)`
- `a = sqrt(a);`

(h) Решето Эратосфена (пользуемся, что: $p_n \approx n \ln n$)

- `for (p = 2; p < n; p++)`
- `if (min_divisor[p] == 0) // is prime`
- `for (x = p + p; x < n; x += p)`
- `if (min_divisor[x] == 0)`
- `min_divisor[x] = p;`

1.2 Домашнее задание

1. (a) Если в определении O опустить условие про N (т.е. оставить просто $\forall n$), будет ли полученное определение эквивалентно исходному?

(b) Тот же вопрос про o .

2. Во всех пунктах нужно ответить на вопрос “за сколько работает”:

(a)

```

• for (i = 0; i < n; i++)
•     if (used[i] == 0)
•         for (j = i; used[j] == 0; j = (j*17+2) % n)
•             used[j] = 1;

```

(b) Школьная арифметика (длины чисел до n , система счисления десятичная):

- i. Сложение в столбик,
- ii. Умножение в столбик,
- iii. Деление в столбик.

(c)

```

• int n = (int) s.length();
• vector<int> pi (n);
• for (int i=1; i<n; ++i) {
•     int j = pi[i-1];
•     while (j > 0 && s[i] != s[j])
•         j = pi[j-1];
•     if (s[i] == s[j]) ++j;
•     pi[i] = j;
• }

```

3. Заполнить табличку.

A	B	O	o	Θ	ω	Ω
$\lg^k n$	n^ϵ					
n^k	c^n					
\sqrt{n}	$n^{\sin n}$					
2^n	$2^{n/2}$					
$n^{\lg m}$	$m^{\lg n}$					
$\lg(n!)$	$\lg(n^n)$					

4. (*) Вот обычный (но медленный) алгоритм Евклида:

```
• int gcd (int x, int y) {  
•     if (x < y)  
•         return gcd(x, y - x);  
•     if (x > y)  
•         return gcd(x - y, y);  
•     if (x == y)  
•         return (x + y) / 2;  
• }
```

Расширим его так:

```
• pair<int,int> gcd+ (int x, int y, int u, int v) {  
•     if (x < y)  
•         return gcd(x, y - x, u, u + v);  
•     if (x > y)  
•         return gcd(x - y, y, u + v, v);  
•     if (x == y)  
•         return pair<int,int> ((x + y) / 2, (u + v) / 2);  
• }
```

Что за пара будет в ответе, при запуске $\text{gcd}+(x, y, x, y)$? (Ответ нужно обосновать)

5. (*) Порой нам нужно найти не только $d = \text{НОД}(x, y)$, но и такую пару целых чисел a и b , что $ax + by = d$. Придумайте, как изменить алгоритм Евклида, чтобы он находил такую пару.

6. Докажите, что следующий алгоритм находит gcd двух чисел:

```
• int gcd(int x, int y) {  
•     if (x == y)  
•         return (x + y) / 2;  
•     if (x % 2 == 0 && y % 2 == 0)  
•         return 2 * gcd(x / 2, y / 2);  
•     if (x % 2 == 0 && y % 2 != 0)  
•         return gcd(x / 2, y);  
•     if (x % 2 != 0 && y % 2 == 0)  
•         return gcd(x, y / 2);  
•     if (x > y)  
•         return gcd (x - y, y);  
•     if (x < y)  
•         return gcd (x, y - x);  
• }
```

Оцените его асимптотику.

2 Практика 2. Жадность. Линейные алгоритмы

1. Дана скобочная последовательность, составленная из скобок '(', ')', '[', ']', '{', '}'. Последовательность называется корректной, если каждой открывающей скобке соответствует закрывающая скобка того же типа, и соблюдается вложенность. Примеры, $(\{ \})$ и $()()$ – корректные, а $()$ и $[()]$ – нет.

Предоставьте алгоритм, который проверяет корректность последовательности за линейное время.

2. Реализовать стек с операциями PUSH, POP, MAX при условии, что каждая операция работает за константное время.
3. Реализовать очередь с помощью двух стеков.
4. Реализовать очередь с операциями PUSH, POP, MAX при условии, что каждая операция работает за константное время.
5. Дано число, представленное n цифрами в десятичной записи без ведущих нулей. Из числа требуется вычеркнуть ровно k цифр так, чтобы результат был максимальным. Задачу требуется решить за линейное от n время.

Внимание! k подается на вход и может быть порядка n . Решение за $O(kn)$ приравнивается к квадратичному от n .

6. Преподаватели сделали n заявок на занятие. Каждое занятие начинается в момент b_i и кончается в момент e_i (занимает интервал $[b_i, e_i)$). Два занятия в одной аудитории быть не могут. Распределите заявки по аудиториям так, чтобы общее число аудиторий было минимально. Решить за $O(n \log n)$.
7. Придумайте алгоритм, который по заданному дереву T на n вершинах строит совершенное паросочетание, или говорит что такого нет. Время $O(n)$.

2.1 Домашнее задание

1. (группа Давыдова) Дано число, представленное n цифрами в десятичной записи без ведущих нулей. Из числа требуется вычеркнуть ровно k цифр так, чтобы результат был максимальным. Задачу требуется решить за линейное от n время.

Внимание! k подается на вход и может быть порядка n . Решение за $O(kn)$ приравнивается к квадратичному от n .

2. Придумайте алгоритм, который по заданному дереву T на n вершинах строит совершенное паросочетание (такое подмножество ребер, что каждая вершина дерева инцидентна ровно одному из них), или говорит, что такого нет. Время $O(n)$.
3. (группа Мишунина) Дан набор кубиков, на каждой грани каждого из них написано целое число. Петя и Вася играют в игру: каждый выбирает себе по кубику, затем каждый бросает свой кубик. Выигрывает тот, у кого выпало число больше (выпадение любой грани у кубика равновероятно). Петя выбирает кубик первым, и ему нужен алгоритм выбора лучшего кубика (лучший кубик — тот, у которого вероятность выиграть не меньше вероятности проиграть против любого другого кубика). Кубики заданы в виде массива шестерок чисел. Напишите псевдокод, возвращающий номер лучшего кубика (любого, если таких несколько) или -1 , если такого кубика не существует (никто не обещает, что набор не пустой...).
4. (группа Давыдова) Дан массив целых чисел a_i . Придумайте структуру данных, которая бы умела отвечать на запросы вида: "По данным l и r вернуть $\sum_{i=l}^r a_i$." за $O(1)$.
Разрешается сделать предподсчет за $O(n)$. (Это означает, что нам дают массив, затем разрешают $O(n)$ времени считать какую-нибудь вспомогательную информацию и только после этого начинают задавать вопросы).
5. Дана последовательность $a_1, a_2, \dots, a_n \in \mathbb{N}$ и $S \in \mathbb{N}$. Найти l, r ($1 \leq l \leq r \leq n$) такие, что сумма $\sum_{i=l}^r a_i = S$. Задачу требуется решить за линейное от n время.
6. Дана последовательность $a_1, a_2, \dots, a_n \in \mathbb{Z}$. Найти l, r ($1 \leq l \leq r \leq n$) такие, что сумма $\sum_{i=l}^r a_i$ была бы максимальной. Задачу требуется решить за линейное от n время.