

# Алгоритмы и структуры данных поиска. Контрольная

Минский ШАД. Весна

20 апреля 2015 г.

1. [2 балла] Дерево на  $n$  вершинах задано своим списком рёбер. На каждом ребре написан вес  $w_{a,b} \in \mathbb{R}$ . Необходимо за  $\mathcal{O}(n)$  предоставить вес самого тяжелого простого пути (вес пути — сумма весов рёбер). Вес пустого пути считается равным нулю.
2. Дан алфавит  $\Sigma$ , причём  $|\Sigma| = n$ . Пусть символы алфавита каким-то образом пронумерованы от 1 до  $n$ . Пусть дана строка  $S$ , тогда полиномиальным хешем от этой строки назовём следующее число:  $h(S) = \left( \sum_{i=1}^{|S|} S_i p^i \right) \bmod M$ , где  $p$  и  $M$  — известные числа. Задача: найти количество таких строк  $S$ , что  $|S| = k$ , а  $h(S) = x$ . Задачу нужно решить для всех  $x = \overline{0 \dots M-1}$ .
  - (a) [1/2 баллов] Решение должно иметь сложность  $\mathcal{O}(kM^2)$ .
  - (b) [1 балл] Решение должно иметь сложность  $\mathcal{O}(\log kM^2)$ .
  - (c) [1 балл] Решение должно иметь сложность  $\mathcal{O}(M \log M \log k)$ .
3. [2 балла] По числу  $n$  предоставьте строку, на которой алгоритм КМП произведёт максимальное количество сравнений, и строку, на которой минимальное. Сколько сравнений проведёт алгоритм в каждом из случаев? Можно считать, что алфавит имеет размер как минимум  $n$ .
4. [2 балла] Дан  $(n, m)$ -граф. Необходимо найти количество способов раскрасить его в два цвета таким образом, чтоб никакие две смежные вершины не были одного цвета. Время работы должно составлять  $\mathcal{O}(n + m)$ .
5. [3 балла] Пусть вы пишете программу для компьютера, которая активно работает с массивом из  $2^k$  машинных слов. Вдруг вы поняли, что вам часто приходится искать максимум на подотрезке этого массива. Вы решили, что стоит использовать sparse-table. Но для реализации этой структуры понадобится  $n = 2^k \times k$  машинных слов. А свободного места в памяти у вас осталось лишь на  $\lfloor \frac{n}{3} \rfloor$  машинных слов. Предложите структуру, которая реализует такую же функциональность как и sparse-table (поиск максимума на любом отрезке за  $\mathcal{O}(1)$ ), но влезет в данное ограничение.
6. [3 балла] Дан алфавит из  $n$  символов. Построить такую строку  $S$  минимальной длины, что все  $n^3$  троек символов из данного алфавита встречаются как подстроки в  $S$ . Сложность алгоритма должна быть линейна от размера получившейся строки.
7. [3 балла] У малыша Пабло есть  $n$  красок (пронумерованных от единицы до  $n$ ) и большое желание раскрасить квадратный холст  $n \times n$  клеток с помощью этих красок. Изначально холст пустой (все клетки имеют цвет ноль). На  $i$ -й день малыш выбирает краску под номером  $c_i$ , которую ещё не выбирал раньше, и закрашивает прямоугольник состоящий из клеток  $(x, y)$ , где  $l_i \leq x \leq r_i$ ,  $d_i \leq y \leq u_i$ . По полученному холсту восстановите любой возможный набор  $(c_i, l_i, r_i, d_i, u_i)$ , который порождает данный холст, за  $\mathcal{O}(n^3)$ .

Например, для следующего холста ответ может быть таким:  $((3, 1, 2, 1, 2), (1, 1, 2, 1, 2), (2, 2, 2, 3, 3))$ :

|   |   |   |
|---|---|---|
| 1 | 1 | 0 |
| 1 | 2 | 2 |
| 0 | 2 | 2 |

8. [2 балла] По числу  $n$  определить  $\bigoplus_{i=0}^n i$  за время  $\mathcal{O}(1)$ , считая, что операция  $\oplus$  (побитовое сложение по модулю два) двух чисел выполняется за  $\mathcal{O}(1)$ .

9. Дан код:

```

1 vector<size_t> just_do_it(const string& s) {
2     vector<size_t> ans(s.size());
3
4     size_t a = 0, b = 0;
5     for (size_t i = 1; i < ans.size(); ++i) {
6         if (b <= i) {
7             a = b = i;
8             while (b < s.size() && s[b] == s[b - a]) ++b;
9             ans[i] = b - a;
10        } else {
11            if (ans[i - a] < b - i) {
12                ans[i] = ans[i - a];
13                continue;
14            }
15            ans[i] = b - i;
16            while (i + ans[i] < s.size() && s[i + ans[i]] == s[ans[i]]) {
17                ++ans[i];
18            }
19            if (i + ans[i] > b) {
20                a = i;
21                b = i + ans[i];
22            }
23        }
24    }
25
26    return ans;
27 }
```

(a) [1 балл] Какой вектор вернёт функция для строки «abacababrestabacaba»?

(b) [1 ½ баллов] Какое значение `ans[50000]` для строки  $a^{10000}b^{10000}c^{20000}a^{10000}b^{30000}c^{12345}$ ? Буква в степени означает, что эта буква повторяется показатель степени раз.

10. [2 балла] Предложите способ реализовать следующий класс:

```

1 class SuperClass {
2 public:
3     void registerValue(int currentTime, int value);
4     int getRecentMin(int currentTime);
5 };
```

«currentTime» — время в секундах от начала отсчёта. Функция «getRecentMin» должна возвращать минимальное зарегистрированное значение за последние 1000 секунд. Можно считать, что для любых двух последовательных вызовов «currentTime» будет не убывать.

| Задание | 1 | 2  | 3 | 4 | 5 | 6 | 7 | 8 | 9  | 10 | Сумма |
|---------|---|----|---|---|---|---|---|---|----|----|-------|
| Баллы   | 2 | 2½ | 2 | 2 | 3 | 3 | 3 | 2 | 2½ | 2  | 24    |