

# Практика по алгоритмам у бакалавров

Сергей Копелиович, Всеволод Опарин

Группа Опарина, Осень, 2014

## 1 Практика 1. Суммы, циклы

### 1.1 Теория

**Арифметическая прогрессия**

- $\sum_{n=a}^b n = \frac{(b+a)(b-a+1)}{2}$
- $\sum_{n=1}^a n = \frac{a(a+1)}{2}$

**Геометрическая прогрессия**

- $\sum_{k=0}^n q^k = \frac{q^{n+1}-1}{q-1}$
- При  $-1 < q < 1$ :  $\sum_{k=0}^{\infty} q^k = \frac{-1}{q-1} = \frac{1}{1-q}$
- Если  $q = \frac{1}{1+\varepsilon}$ :  $1 - q = \frac{\varepsilon}{1+\varepsilon} \approx \varepsilon \Rightarrow \sum_{k=0}^{\infty} (1 + \varepsilon)^{-k} \approx \frac{1}{\varepsilon}$

**Сумма гармонического ряда**

- $1 + \lfloor \log_2 n \rfloor \geq \underbrace{\frac{1}{1} + \frac{1}{2} + \frac{1}{2}}_1 + \underbrace{\frac{1}{4} + \frac{1}{4} + \frac{1}{4} + \frac{1}{4}}_1 + \underbrace{\frac{1}{8} + \dots}_{1 \dots} \geq \sum_{k=1}^n \frac{1}{k} = \frac{1}{1} + \frac{1}{2} + \frac{1}{3} + \frac{1}{4} + \frac{1}{5} + \frac{1}{6} + \frac{1}{7} + \frac{1}{8} + \dots \geq \frac{1}{1} + \underbrace{\frac{1}{2}}_{1/2} + \underbrace{\frac{1}{4} + \frac{1}{4}}_{1/2} + \underbrace{\frac{1}{8} + \frac{1}{8} + \frac{1}{8} + \frac{1}{8}}_{1/2} + \dots \geq 1 + \frac{1}{2} \lfloor \log_2 n \rfloor \Rightarrow \sum_{k=1}^n \frac{1}{k} = \Theta(\log n)$

## Про интегралы

- Формула Ньютона-Лейбница:  $\int_a^b f'(x)dx = f(b) - f(a)$
- Пример:  $\ln'(n) = \frac{1}{n} \Rightarrow \int_1^n \frac{1}{x} dx = \ln n - \ln 1 = \ln n$
- $\int_a^b \left[ \min_{y \in [a..b]} f(y) \right] dx \leq \int_a^b f(x) dx \leq \int_a^b \left[ \max_{y \in [a..b]} f(y) \right] dx$
- Для монотонно убывающей  $f(x)$  имеем  $f(a) \geq \int_a^{a+1} f(x) dx \geq f(a+1)$ .
- Для монотонно возрастающей  $f(x)$  имеем  $f(a) \leq \int_a^{a+1} f(x) dx \leq f(a+1)$ .
- Пусть  $X(n) = \sum_{i=1}^n \frac{1}{i}$ , значит,  $X(n) - \frac{1}{n} = \sum_{i=1}^{n-1} \frac{1}{i} \geq \int_1^n \frac{1}{x} dx = \ln n \geq \sum_{i=2}^n \frac{1}{i} = X(n) - 1 \Rightarrow \ln n + \frac{1}{n} \leq X(n) \leq \ln n + 1 \Rightarrow X(n) = \ln n + \mathcal{O}(1)$

## 1.2 Класс

### Задачи про суммы

1.  $\sum_{k=0}^{\infty} \frac{n}{2^k}$
2.  $\sum_{k=0}^{\infty} \frac{n}{1.01^k}$
3.  $\sum_{k=1}^n \frac{1}{2^{k-1}}$
4.  $\prod_{k=2}^n (1 - k^{-2})$
5.  $\prod_{k=1}^n (2 \cdot 4^k)$

**Задачи про циклы for.** Во всех задачах нужно ответить на вопрос “за сколько работает”:

1. Найти такие  $a, b, c$ :  $abc = N$ ,  $a + b + c = \min$ . Решение:

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

2. Поиск делителей без деления

```
1. b = N;
2. for (a = 1; a * a <= N; a++) {
3.     while (a * b > N)
4.         b--;
5.     if (a * b == N)
6.         printf("%d %d\n", a, b);
7. }
```

3. Еще одно решение задачи (1)

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

Получить  $\mathcal{O}(N^{5/6})$

4. (\*) И еще одно решение задачи (1)

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

Получить  $\mathcal{O}(N^{2/3})$

### 1.3 Домашнее задание

Дедлайн: 15 сентября, 23.59

Задачи на цикл `for`. Оцените время работы.

1. Про строки

```
1. for (a = 1; a < n; a++)
2.     for (b = 0; b < n; b += a)
3.         ;
```

## 2. Partition

```
1. a = 1, b = n;
2. while (a < b) {
3.     while (x[a] < M && a <= b) a++;
4.     while (x[b] > M && a <= b) b--;
5.     if (a <= b) swap(x[a++], x[b--]);
6. }
```

Дополнительный вопрос: что делает этот код?

## 3. Логарифмы

```
1. while (a >= 2)
2.     a = sqrt(a);
```

Дополнительный вопрос: а если бы вместо 2 было бы 1?

## 4. Решето Эратосфена

```
1. for (p = 2; p < n; p++)
2.     if (min_divisor[p] == 0) // is prime
3.         for (x = p + p; x < n; x += p)
4.             if (min_divisor[x] == 0)
5.                 min_divisor[x] = p;
```

(\*) Доказать  $\mathcal{O}(N \log \log N)$  (пользуемся не доказанным фактом:  $p_n \approx n \ln n$ )

## 5. Перестановки и циклы

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

## Суммы

1.  $\sum_{k=1}^{\infty} \frac{k}{2^k}$

2.  $\sum_{k=1}^{\infty} \frac{k^2}{2^k}$

3. Докажите, что  $\sum_{k=1}^n \frac{1}{k^{3/2}} = \mathcal{O}(1)$

## 2 Практика 2. Элементарные структуры данных

### 2.1 Класс

1. Дана последовательность чисел  $a_1, a_2, \dots, a_n$  и число  $S$ . Все числа положительные целые. Требуется за линейное время найти подотрезок последовательности такой, что сумма его элементов  $\sum_{i=l}^r a_i = S$ . Время –  $O(n)$ .
2. Дана последовательность чисел  $a_1, a_2, \dots, a_n$ . Все числа целые. Требуется за линейное время найти подотрезок последовательности такой, что сумма его элементов  $\sum_{i=l}^r a_i$  максимальна. Время –  $O(n)$ .
3. Дана последовательность чисел:  $x_1 = a$ ,  $x_{i+1} = f(x_i)$ , найти с  $O(1)$  дополнительной памяти длину периода  $T$  и предпериода  $L$  за время  $O(L + T)$ .
4. Частичные суммы:
  - (a) Много раз сделать += на отрезке, в конце один раз вывести массив.
  - (b) Сперва много раз += на отрезке, затем много раз “сумма на отрезке”.
  - (c) В каждой целой точке  $x$  числовой прямой есть  $f[x]$ , изначально равная нулю. Те же запросы, что и в предыдущем пункте. Координаты запросов целые от 0 до  $10^{18}$ .

### 2.2 Домашнее задание

Дедлайн: 22 сентября, 23.59

1. Задачи на стек.
  - (a) В массиве найти для каждого элемента ближайший меньший среди соседей слева и справа за  $O(n)$ .
  - (b) В последовательности  $a_1, a_2, \dots, a_n$  найти подотрезок, на котором максимизируется значение  $(r - l + 1) \cdot \min_{i \in [l, r]} a_i$ , за  $O(n)$ .
  - (c) Дана матрица из нулей и единиц. Найти наибольший по площади подпрямоугольник, состоящий только из нулей за  $O(n^2)$ .
2. Дан массив из  $2n$  чисел. Найти минимальное **И** максимальное за  $3n - 2$  сравнения.
3. Для этих задач приведите код на C++. Время работы алгоритма должно быть линейно. Множество и мультимножество можно хранить в виде отсортированного массива. Даны два множества  $A$  и  $B$  в отсортированном виде, за  $O(|A| + |B|)$  построить в таком же виде их
  - (a) пересечение. Пример:  $\{1, 2, 3\} \cap \{2, 4\} = \{2\}$ .

(b) разность. Пример:  $\{1, 2, 3\} \setminus \{2, 4\} = \{1, 3\}$ .

(c) объединение. Пример:  $\{1, 2, 3\} \cup \{2, 4\} = \{1, 2, 3, 4\}$ .

P.S. Если вы пишете в  $\text{\LaTeX}$ , попробуйте заодно пакет `\usepackage{minted}`. В `dropbox` Сережи Копелиовича есть пример. За подробной справкой можно обращаться к нему и Диме Лапшину.

## 3 Практика 3. Сортировки

### 3.1 Класс

1. Даны два массива  $a$  и  $b$  длины  $n$ , сгенерировать все попарные суммы  $a_i + b_j$  в отсортированном порядке.
  - (a) За  $\mathcal{O}(n^2 \log n)$ .
  - (b) За  $\mathcal{O}(n^3)$  с использованием  $\mathcal{O}(n)$  дополнительной памяти.
  - (c) За  $\mathcal{O}(n^2 \log n)$  с использованием  $\mathcal{O}(n)$  дополнительной памяти.
2. В свободное время Анка-пулеметчица любит сортировать патроны по серийным номерам. Вот и сейчас она только разложила патроны на столе в строго отсортированном порядке. Но тут Иван Васильевич распахнул дверь с такой силой, что все патроны на столе подпрыгнули и немного перемешались. Оставив ценные указания, Иван Васильевич отправился восвояси. Как оказалось, патроны перемешались не сильно. Каждый патрон отклонился от своей позиции не более чем на  $k$ . Всего патронов  $n$ . Помогите Анке отсортировать патроны.
  - (a) Отсортируйте патроны за  $\mathcal{O}(nk)$ .
  - (b) Отсортируйте патроны за  $\mathcal{O}(n + I)$ , где  $I$  — число инверсий.
  - (c) Докажите нижнюю оценку на время сортировки  $\Omega(n \log k)$ .
  - (d) Отсортируйте патроны за  $\mathcal{O}(n \log k)$ .
3. Дан массив из  $n+1$  целого числа от 1 до  $n$ . Массив доступен только на чтение, есть  $\mathcal{O}(1)$  дополнительной памяти. Найти за  $\mathcal{O}(n)$  любое число, которое встречается хотя бы два раза.
4. Дано  $n$  точек на плоскости. Соединить их
  - (a)  $(n-1)$ -звенной ломаной без самопересечений (не замкнутой);
  - (b)  $n$ -звенной ломаной без самопересечений (замкнутой)за  $\mathcal{O}(n \log n)$ .
5. Дан набор из  $N$  отрезков  $[a_i, b_i]$ . Числа  $a_i, b_i$  — вещественные.
  - (a) Найти такое вещественное  $x$ , что  $|\{i : x \in [a_i, b_i]\}|$  максимально.
  - (b) Длину объединения отрезков.
  - (c) Для каждого  $k$  посчитать, сколько точек на прямой покрыто ровно  $k$  отрезками.
6. Даны два массива из положительных чисел  $a$  и  $b$ .  $|a| = |b|$ . Выбрать массив  $p$ :  $k$  различных чисел от 1 до  $n$  так, чтобы:

- (a)  $\sum_{i=1}^n a_{p_i} b_{p_i} \rightarrow \max.$   
 (b)  $\frac{\sum_{i=1}^n a_{p_i}}{\sum_{i=1}^n b_{p_i}} \rightarrow \max.$   
 (c)  $(\sum_{i=1}^n a_{p_i})(\sum_{i=1}^n b_{p_i}) \rightarrow \max.$

## 3.2 Домашнее задание

Дедлайн: 29 сентября, 23.59

- Даны два массива  $a$  и  $b$  одинаковой длины.  
Нужно найти такую перестановку  $p$ , что  $\sum_{i=1}^n a_{p_i} b_i \rightarrow \max.$  Решение обосновать.
- Рассмотрим бинарную кучу, у которой дерево необязательно сбалансированно.  
Пусть  $d(v)$  – расстояние вниз от вершины  $v$  до ближайшего листа, а  $size(v)$  – число вершин в поддереве с корнем в вершине  $v$ .  
Куча называется левацкой, если  $\forall v: d(l[v]) \geq d(r[v]).$ 
  - докажите, что  $size(v) \geq 2^{d(v)}$
  - напишите быстрый Merge для данной кучи, дайте оценку времени работы
- Дано  $2 \cdot n - 1$  коробок с черными и белыми шарами. В  $i$ -ой коробке находится  $w_i$  белых и  $b_i$  черных шаров. Всего в коробках находится  $W$  белых и  $B$  черных шаров. Требуется выбрать  $n$  коробок, чтобы суммарное число белых шаров в них было не менее  $\frac{W}{2}$ , а черных не менее  $\frac{B}{2}$ . Решить за  $\mathcal{O}(n \log n)$ .
- На прямой расположено  $n$  точек  $p_1, p_2, \dots, p_n$  в порядке возрастания. Каждая точка имеет вес  $w_i \geq 0$ . Требуется найти такую точку  $q$ , что  $\sum_i w_i \cdot |p_i - q|$  имела бы минимальное значение. Время работы:  $\mathcal{O}(n)$ .

**Примечание.** Точки на вход подаются в порядке возрастания.



## 4 Практика 4. Куча и медиана

### 4.1 Класс

1. Пусть есть структура данных, реализующая интерфейс `PriorityQueue (Add, DeleteMin)`. Как расширить эту структуру данных операциями `Build`, `Delete`, `DecreaseKey`, `Merge`?
2. Робот Иван Семеныч пробует пирожки. Содержимое пирожков делится на три типа. Всего пирожков  $n$ . Каждый пирожок можно попробовать не более одного раза. Пирожки можно менять местами. Память у робота маленькая,  $O(\log n)$  бит. Помогите Ивану Семенычу отсортировать пирожки по типу: сначала первый, потом второй, потом третий. Сортировка должна работать за линейное время.
3. Найти отрезок массива, на котором  $(\min_{i \in [l, r]} a_i) \cdot \sum_{i \in [l, r]} a_i$  максимально. Время  $O(n)$ .
  - (a) Все числа положительны.
  - (b) Числа целые, 32-битные. Решение с использованием минимума на отрезке за  $O(1)$ .
  - (c) Числа целые, 32-битные. Простое решение.
4. Дана обычная бинарная куча, за сколько можно узнать  $k$ -й минимум?
  - (a)  $O(k \log n)$
  - (b)  $O(k^2)$
  - (c)  $O(k \log k)$
5. Оцените число сравнений, которое сделают
  - (a) `MergeSort`, если в массиве ровно  $n = 2^k$  элементов.
  - (b) `HeapSort`, если в массиве ровно  $n = 2^k - 1$  элементов.
  - (c) `QuickSort`, если в массиве все элементы различны, а выбирается случайный.

Во всех пунктах нужны точные константы.

### 4.2 Домашнее задание

#### Дедлайн: 6 октября, 23.59

Если не успели рассказать.  $k$ -ая порядковая статистика в массиве  $a$  является  $k$ -ым по порядку элементом. Т.е. если массив  $a$  отсортировать, то это будет  $a_k$ .

Во всех задачах ниже можно считать, что существует процедура поиска  $k$ -ой статистики за  $O(n)$ , где  $n$  – длина массива.

1. Медианой называется  $\lfloor \frac{n}{2} \rfloor$ -я порядковая статистика. Придумайте структуру данных на основе `heap`, которая умеет делать `Insert(x)`, `DeleteMedian()`, все операции за  $\mathcal{O}(\log n)$ .
2. Дан массив `A[1..n]` из  $n$  различных чисел. Массив не обязательно отсортирован. Требуется найти  $k$  ближайших к медиане элементов за линейное время. Решить для двух метрик.

(a) По позиции в отсортированном массиве.

$$d(x, \text{median}) = |\text{pos}(x) - \text{pos}(\text{median})|,$$

где  $\text{pos}(x)$  — позиция элемента  $x$  в отсортированном массиве.

(b) По значению.

$$d(x, \text{median}) = |x - \text{median}|.$$

3. Дано два массива  $a_1, a_2, \dots, a_n$  и  $p_1, p_2, \dots, p_m$ . Второй массив отсортирован. Нужно за  $\mathcal{O}(n \log m)$  для каждого  $i$  найти  $p_i$ -ую порядковую статистику в массиве  $a$ .

## 5 Практика 5. Вероятности и бин. поиск

### 5.1 Класс

1. Пусть  $X$  – число выпадений орла после  $n$  бросаний фальшивой монетки. Докажите, что  $\mathbf{D}[X] = npq$ .

### 5.2 Домашнее задание

Дедлайн: 13 октября, 23.59

1. Бинарный поиск на массиве

- (a) Выразить `upper_bound` для целых чисел через `lower_bound`.
- (b) Докажите, что нельзя сделать и `lower_bound`, и `upper_bound` одновременно, используя в худшем случае меньше чем  $2 \log_2 n + \mathcal{O}(1)$  сравнений?
- (c) Сделать предподсчет за  $\mathcal{O}(n \log n)$ , чтобы за  $\mathcal{O}(\log n)$  отвечать на запрос “сколько раз число  $x$  встречается на отрезке  $[l..r]$ ”?

2. Под  $\mathbb{F}_2$  будем подразумевать поле из двух элементов: 0 и 1. Определим для них операции сложения и умножения как обычные по модулю 2. Например,  $1 + 1 = 0$ .  $\mathbb{F}_2^n$  – это вектор размерности  $n$  над полем  $\mathbb{F}_2$ .

Определим  $U(S)$  как равномерное распределение над множеством  $S$ . Т.е. каждый элемент  $S$  может выпасть с вероятностью  $\frac{1}{|S|}$ .

- (a) Докажите для любого ненулевого  $a \in \mathbb{F}_2^n$  что

$$\Pr_{x \leftarrow U(\mathbb{F}_2^n)} [a \cdot x = 0] = \frac{1}{2}$$

Примечание.  $a \cdot x = \sum_{i=1}^n a_i \cdot x_i$  по модулю 2.

- (b) Даны три матрицы  $A$ ,  $B$  и  $C$  размерности  $n \times n$  над полем  $\mathbb{F}_2$ . Постройте алгоритм, который работает  $\mathcal{O}(n^2)$  и
  - возвращает 1, если  $A \cdot B = C$  с вероятностью 1;
  - возвращает 0, если  $A \cdot B \neq C$  с вероятностью не меньше  $\frac{1}{2}$ .

3. Есть три контейнера. В одном из них лежит золотой шар. Оба других пусты. Разрешается открыть только один контейнер. Вы сели на первый попавшийся, и в этот момент открылся второй. Он оказался пуст. Есть ли теперь разница, какой контейнер открывать? Ответ обосновать.

## 6 Практика 6. Динамика

### 6.1 Домашнее задание

Дедлайн: 20 октября, 23.59

1. На билете есть  $2n$  значный номер. Билет считается счастливым, если сумма первых  $n$  цифр совпадает с суммой последних  $n$  цифр. По заданому числу  $n$  требуется найти число счастливых  $2n$  значных билетов за  $O(n^2)$ . Считать, что стандартные арифметические операции над числами выполняются за  $O(1)$ .
2. Судно атакуют пираты. Для каждого пирата известны его азимут  $a_i$  и время  $t_i$ , через которое пират приплывет и совершит непотребство. Однако, у судна есть лазерная пушка, которой оно защищается. У пушки есть начальный азимут  $a$  и угловая скорость вращения  $\omega$ . Пушка уничтожает все объекты, на которые она сейчас направлена. Помогите судну определить порядок уничтожения пиратов за  $O(n^2)$ , чтобы не допустить непотребства.
3. Дан текст из слов длины  $l_1, l_2, \dots, l_n$ . Порядок слов менять нельзя. Хотим сделать красивый абзац. В красивом абзаце длина каждой строки ровно  $L$  (остаток заполняется пробелами). Между каждой парой слов стоит хотя бы один пробел. Обозначим число пробелов в строке  $i$  через  $gap_i$ . Нужно сделать абзац с минимальным значением  $\sum_i gap_i^3$ , где сумма пробегается по всем строкам построенного абзаца. Решить за  $O(n \cdot L)$ .
4. Есть три контейнера. В одном из них лежит золотой шар. Оба других пусты. Разрешается открыть только один контейнер. Вы сели на первый попавшийся. Известно, что в этот момент откроется один из двух других. Если в одном из них есть золотой шар, то второй открывается с вероятностью  $p$ . Открылся пустой. Какова вероятность, что в контейнере, на котором вы сидите, есть золотой шар? Решить в общем случае, при  $p = 1$  и  $p = \frac{1}{2}$ .

## 7 Практика 7. Динамика

### 7.1 Домашнее задание

Дедлайн: 27 октября, 23.59

1. Скобочная последовательность почти правильная, если является префиксом некоторой правильной скобочной последовательности. Найти число почти правильных последовательностей длины  $n$ . Решить через динамику за квадрат и написать формулу в явном виде.
2.  $k$ -ичная система исчисления – это система, в которой присутствует  $k$  цифр. Найдите количество  $n$ -значных чисел в  $k$ -ичной системе, не содержащих двух нулей подряд за  $O(n)$ . Можно считать, что ответ считается по модулю маленького простого числа, поэтому все стандартные арифметические операции выполняются за  $O(1)$ .
3. Шаблон для строки состоит из символов латинского алфавита, ? и \*. Мы говорим, что строка  $s$  подходит под шаблон  $p$ , если существует каждый ? в шаблоне можно заменить на букву, а \* на возможно пустую строку из латинских букв так, что результат будет равен  $s$ . Для строки  $s$  и шаблона  $p$  определите, подходит ли строка под шаблон за  $O(|s| \cdot |p|)$ .
4. Взвешанный бинарный поиск элемента  $x$ . Бинарный поиск взвешанный, если стоимость обращения к  $i$ -му элементу массива равна  $c_i > 0$ . Мы не знаем сам массив, не знаем  $x$ , который будем искать, знаем только массив стоимостей  $c$ . Нужно за  $O(n^3)$  найти стоимость поиска элемента в худшем случае. Внутри бинарного поиска не обязательно сравнивать с серединой, мы сами выбираем элемент, с которым сравнивать.
5. Дана строка  $s$  из цифр 0 и 1. Строка свободна от  $s$  если не содержит  $s$  в качестве подстроки. Требуется найти число строк, свободных от  $s$ , длины  $n$  из цифр 0 и 1. Можно считать, что ответ считается по модулю маленького простого числа, поэтому все стандартные арифметические операции выполняются за  $O(1)$ . Время:
  - (a)  $O(|s|^3 + |s| \cdot n)$ ;
  - (b)  $O(|s| \cdot n)$ .

## 8 Практика 8. Динамика

### 8.1 Практика

#### 1. Операции с битами

- (a) Дано  $w$ -битное число, за  $\mathcal{O}(1)$ , проверить, является ли число степенью двойки.
- (b) Дано  $w$ -битное число, за  $\mathcal{O}(1)$ , проверить, правда ли, что в битовой записи никакие две единицы не идут подряд.
- (c) Дано  $w$ -битное число, за  $\mathcal{O}(\log w)$  найти старший единичный бит.
- (d) Дано  $w$ -битное число, за  $\mathcal{O}(\log w)$  посчитать количество единичных бит.
- (e) Дано  $w$ -битное число, найти младший единичный бит быстрее чем  $\mathcal{O}(w)$ .

#### 2. Задачи про паросочетания в произвольном графе

- (a) Посчитать количество паросочетаний
- (b) Найти максимальное по весу паросочетание

### 8.2 Домашнее задание

**Дедлайн: 10 ноября, 23.59**

1. Для каждого множества вершин в графе посчитать количество независимых подмножеств. а)  $\mathcal{O}(3^n)$ ; б)  $\mathcal{O}(2^n \cdot n)$ ; в)  $\mathcal{O}(2^n)$ .
2. Дан двудольный граф, в первой доле  $m \leq 15$  вершин, во второй  $n \leq 1000$  вершин. Предложите алгоритм, который считает количество паросочетаний, покрывающих все  $m$  вершин первой доли.
3. Дан неориентированный граф. Посчитать за  $\mathcal{O}^*(2^n)$  количество способов все вершины графа разбить на циклы. Каждая вершина должна лежать ровно в одном цикле. Разбиения на циклы различны, если различны множества использованных ребер. ( $\mathcal{O}^*$  – это как  $\mathcal{O}$ , только вместо константы может быть полином).
4. Васе дали кость на  $d$  граней. На гранях написаны числа от 1 до  $d$ . Каждая грань выпадает равновероятно. Вася может кинуть кость не более  $n$  раз. Выигрыш Васи – результат последнего броска. Вася желает выиграть как можно больше. Найти мат. ожидание выигрыша, если Вася играет оптимально. Время  $\mathcal{O}(n)$ .

## 9 Практика 9. Поиск в глубину

### 9.1 Практика

1. Дан граф  $G = \langle V, E \rangle$  и два множества вершин  $A$  и  $B$ . За  $O(V + E)$  проверить, есть ли путь из какой-нибудь вершины  $a \in A$  в какую-нибудь вершину  $b \in B$ .
2. Дан взвешенный неориентированный граф  $G = \langle V, E \rangle$  и два множества вершин  $A, B \subseteq V$ . Найти путь из  $A$  в  $B$  такой, что вес максимального ребра пути минимален за  $O((V + E) \log E)$ .
3. Дан граф  $G = \langle V, E \rangle$ . Найти цикл через данное ребро  $e \in E$  за  $O(E)$ .
4. Дан граф  $G = \langle V, E \rangle$ . Найти цикл через данную вершину  $v \in V$  за  $O(E)$ .
5. Найти цикл в неориентированном графе  $G = \langle V, E \rangle$  за  $O(V)$ .
6. Дан неориентированный граф  $G = \langle V, E \rangle$ . Требуется ориентировать его ребра так, чтобы он стал сильно связным.
7. Дан связный граф  $G = \langle V, E \rangle$ . Требуется добавить в него минимальное число ребер, чтобы он стал эйлеровым (содержал эйлеров цикл \setminus путь).
8. Дан неориентированный граф  $G = \langle V, E \rangle$ . Требуется покрыть его ребра реберно-непересекающимися путями. Найти покрытие с минимальным числом таких путей.
9. Дан неориентированный граф  $G = \langle V, E \rangle$ . Требуется найти в нем  $T$  циклов так, чтобы каждый цикл содержал уникальное ребро.  $T$  должно быть максимальным.

## 9.2 Домашнее задание

Дедлайн: 17 ноября, 23.59

1. Разбить множество вершин графа на две доли так, чтобы у каждой вершины был сосед в другой доле.
2. Есть комнаты и двери между комнатами. Нужно каждую дверь покрасить с одной стороны в зеленый цвет, с другой в оранжевый цвет так, чтобы для каждой комнаты количества зеленых и оранжевых дверей отличались не более чем на один.  $O(V + E)$ .
3. Найти гамильтонов цикл на графе бинарных строк длины  $n$ . Ребро между строками есть, если  $n - 1$ -суффикс первой совпадает с  $n - 1$ -префиксом второй.
4. У каждой вершины не более 3 врагов. Разбить на 2 доли так, чтобы с вершиной в долю попало не более 1 врага.  $O(V + E)$ .
5. Дан  $2^k$ -регулярный граф (все степени равны  $2^k$ ). Нужно разбить все его ребра на паросочетания за  $O(kE)$ .