# Майская проектная смена по математике и теоретической информатике

Сириус, 2024

МКН СПбГУ

# Коммуникационные игры

**Аннотация**

Проект посвящён изучению коммуникационной сложности и её применений в различных областях компьютерных наук. Основной объект изучения — это игра двух игроков, Алисы и Боба, живущих в разных городах, в которой они должны вычислить значение некоторой функции $f(x,y)$, где $x$ известен только Алисе, $y$ — только Бобу. Игрокам разрешено общаться между собой, посылая друг другу битовые сообщения. Их задача — вычислить $f(x,y)$, передав как можно меньше сообщений. Коммуникационная сложность естественным образом возникает в потоковых и распределённых алгоритмах, схемной сложности и сложности доказательств, и в других областях компьютерных наук. Как это часто бывает в теоретической информатике, задачи, которые будут у нас возникать, имеют очень простые формулировки, но интересные и совсем нетривиальные доказательства, поэтому в течение смены нам предстоит освоить множество техник и трюков.

Мы начнем с классической коммуникационной сложности, далее рассмотрим ее различные модификации, такие как полудуплексная коммуникационная сложность, коммуникационная сложность с оракулом и универсальные коммуникационные протоколы, которые возникают в некоторых задачах теоретической информатики, и дойдем до множества открытых задач.

**Белова Татьяна Сергеевна (ПОМИ РАН)**
Аспирант и младший научный сотрудник ПОМИ РАН, выпускница МКН СПбГУ. Лектор курса по высокоточной сложности в СПбГУ 2024-2025, вела практические занятия по дискретной математике и алгоритмам в СПбГУ, ИТМО, ВШЭ. Победитель конкурса индивидуальных грантов «Молодая математика России». Финалист ACM ICPC.

**Игнатьев Артур Андреевич (МКН СПбГУ, ВШЭ)**
Студент магистратуры «Разработка программного обеспечения и науки о данных», выпускник бакалавриата МКН СПбГУ. Младший научный сотрудник лаборатории теории игр ВШЭ. На МКН ведет практики по курсу «Теоретическая информатика».

**Дементьев Юрий Ильич (МКН СПбГУ, ВШЭ)**
Студент магистратуры «Разработка программного обеспечения и науки о данных», выпускник бакалавриата МКН СПбГУ. Исследователь лаборатории теории игр ВШЭ, аналитик в MY.GAMES. На МКН ведет практики по курсу «Теоретическая информатика».

# Содержание

# 1 Игра Алисы и Боба

**Задача 1.0** (0 баллов). *Алиса и Боб играют в следующую игру. Они находятся в разных городах, Алисе сообщается число $x$, а Бобу — число $y$, причём $x$ и $y$ — это 0, 1, или 2. В их распоряжении есть устройство связи, которое позволяет передавать друг другу битовые сообщения (т.е. за одно сообщение можно послать «0» или «1»). Алиса и Боб могут заранее договориться о том, какие сообщения они будут посылать. Как им договориться, чтобы в результате оба игрока узнали значение $x + y$?*

**Разбор задачи 1.0.** Так как $x$ и $y$ принимают только значения 0, 1, или 2, то результат сложения $x + y$ принимает целые значения от 0 до 4. Алиса и Боб могут, например, использовать следующий подход: Алиса посылает Бобу двоичную запись $x$, Боб вычисляет $z = x + y$ и посылает Алисе двоичную запись $z$. Давайте посчитаем, сколько сообщений потребуется передать игрокам в худшем случае. Чем больше число, тем длиннее его двоичная запись, поэтому худший случай будет достигаться на входе $(2, 2)$. В этом случае Алиса передаст Бобу два сообщения «1» и «0», задающие двоичную запись числа 2, а Боб перешлёт Алисе три сообщения «1», «0», «0», задающие двоичную запись числа 4. Итого потребуется 5 сообщений (см. рис. 1).

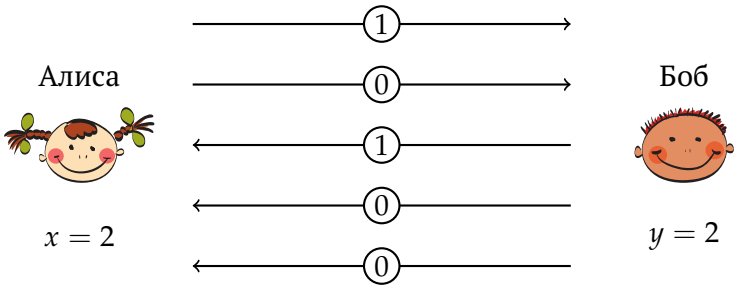



Рис. 1. Возможное взаимодействие Алисы и Боба на входе $(2, 2)$ в задаче 1.0.

Обсудим, как Алиса и Боб будут вести себя на других входах. Пусть $x = 1$, а $y = 0$. По договорённости Алиса должна послать Бобу $x$. Она может попробовать сделать это, послав одно сообщение «1», но это приведёт к неопределённости — Боб не будет знать, следует ли ждать от Алисы ещё сообщения (например, если она собирается передать $x = 2$), или Алиса уже закончила и $x = 1$. Поэтому их общение должно быть устроено так, чтобы таких неоднозначностей не возникало. Например, Алиса и Боб могут договориться, что Алиса всегда посылает два бита: «0» и «0» при $x = 0$, «0» и «1» при $x = 1$, и «1» и «0» при $x = 2$. Боб в свою очередь всегда отвечает, посылая три бита, кодирующие числа от 0 до 4 аналогичным образом.

**Задача 1.1** (1 балл). *Как решить задачу 1.0 так, чтобы Алиса и Боб в сумме послали не более четырёх сообщений?*

**Задача 1.2** (1 балл). *Как решить задачу 1.0 так, чтобы Алиса и Боб всегда посылали не более четырёх сообщений, но при этом на некоторых входах посылали строго менее четырёх сообщений?*

**Задача 1.3** (3 балла). *Докажите, что не существует способа решить задачу 1.0, при котором Алиса и Боб будут всегда посылать не более трёх сообщений.*

Последняя задача принципиально отличается от первых трёх, т.к. в ней нужно не придумать какой-то способ, а *доказать*, что такого способа не существует. Решением этой

задачи должно быть (математически строгое) доказательство. Давайте для примера разберём доказательство более простого утверждения «не существует способа решить задачу 1.0, посылая не более двух сообщений». Сначала рассмотрим пример *некорректного* рассуждения.

> За два сообщения Алиса сможет передать Бобу значение $x$, но ведь Бобу тоже нужно что-то послать в ответ Алисе, поэтому двух сообщений не хватит.

Это рассуждение не является корректным доказательством, т.к. мы рассуждаем про какой-то конкретный способ решения, который используют Алиса и Боб. Нам же нужно доказать, что *никакой* способ решения не сработает. Корректное доказательство могло бы выглядеть, например, так.

> Если при каких-то $x$ и $y$ все сообщения посылает один из игроков, то сам этот игрок не получает никаких сообщений, а, следовательно, он никак не может восстановить значение $x+y$ — этот игрок знает только одно из слагаемых, и не знает ничего про второе. Поэтому всегда сообщения посылают оба игрока. Рассмотрим поведение игроков на парах входов $(0,0)$, $(1,0)$ и $(2,0)$. Во всех трёх случаях Боб получает только одно битовое сообщение. Это значит, что на каких-то двух входах Боб получает одно и то же сообщение. Следовательно, две пары входов с разным $x$ и одинаковым $y$ не отличимы с точки зрения Боба, поэтому он не может знать $x+y$ — если бы он знал, то смог бы восстановить $x$.

**Задача 1.4** (1 балл). *Пусть Алиса и Боб вместо суммы хотят вычислить произведение двух целых чисел от* 0 *до* $n = 2^k - 1$. *Как это сделать за* $2k$ *сообщений?*

Для продолжения нам нужно немного поговорить о множествах, функциях и деревьях.

## 2 Множества, функции и деревья

- *Множества* состоят из *элементов*. Запись $x \in M$ означает, что $x$ является элементом множества $M$. Множество можно задать, перечислив его элементы в фигурных скобках, например, $M = \{0,1,2,3\}$, или при помощи записи с условием, $A = \{x \mid [\textit{условие}]\}$ (множество таких $x$, для которых верно [*условие*]). Например, $\{x \mid x \in \mathbb{Z}, x \bmod 2 = 0\}$ — множество чётных целых чисел.

- Множество *конечно*, если оно содержит конечное число элементов. *Мощность* конечного множества $A$ — это количество элементов в нём, обозначается $|A|$.

- Множество $A$ является *подмножеством* множества $B$ (записывается как $A \subset B$), если все элементы $A$ являются элементами $B$.

- *Декартово произведение* $A \times B$ состоит из всех возможных упорядоченных пар $(a,b)$, где $a \in A$ и $b \in B$.

  $$A \times B = \{(x,y) \mid x \in A \text{ и } y \in B\}$$

  Декартово произведение множества $A$ на само себя удобно записывать так:

  $$A \times A = A^2, \quad \underbrace{A \times A \times \cdots \times A}_{n} = A^n.$$

  Например, $\{0,1\}^5$ — множество всех битовых строк длины 5.

- Подмножество $R$ множества $A \times B$ называют *отношением* между множествами $A$ и $B$.
- Отношение $f \subset A \times B$ называют *функцией из $A$ в $B$*, если оно не содержит пар с одинаковым первым членом и разными вторыми. Это означает, что для каждого $a \in A$ существует не более одного $b \in B$, при котором $(a, b) \in f$. Множество всех $a \in A$, для которых существует такое $b \in B$, что $(a, b) \in f$, называется *областью определения* $f$. Для всех $a$ из области определения $f$ можно определить *значение* $f$ на аргументе $a$ как тот единственный элемент $b \in B$, при котором $(a, b) \in f$ (обозначается $f(a)$). Если область определения $f$ совпадает с $A$, то пишут: $f\colon A \to B$ (т.е. $f$ сопоставляет каждому элементу $A$ элемент $B$).

В соответствии с этими определениями мы можем рассматривать сложение двух целых чисел 0 до 2 как функцию из множества $\{0,1,2\} \times \{0,1,2\}$ в множество $\{0,\dots,4\}$. Таким образом, в задаче 1.0 мы рассматривали функцию от двух аргументов

$$f\colon \{0,1,2\} \times \{0,1,2\} \to \{0,1,2,3,4\}$$

такую, что $f(x,y) = x + y$. Подробнее о множествах и функциях можно прочитать здесь.

*Упорядоченное корневое двоичное дерево* состоит из *вершин*, соединённых *рёбрами* следующим образом (см. рис. 2, вершины обозначаются кружочками, а рёбра — отрезками):

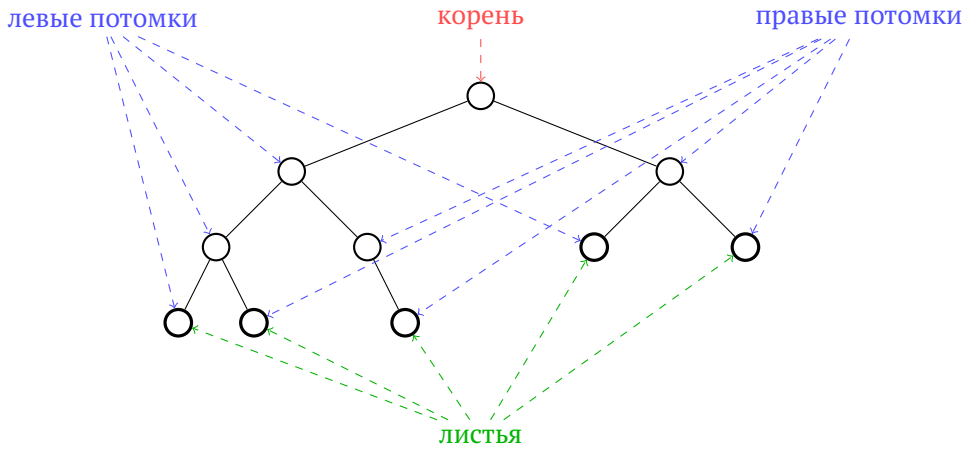



Рис. 2. Упорядоченное корневое двоичное дерево.

- есть выделенная вершина — *корень*,
- у каждой вершины не более двух *потомков*, с которыми она соединена рёбрами,
- корень — это единственная вершина, которая не является является чьим-то потомком,
- каждый потомок является либо *левым*, либо *правым*, если потомка два, то один из них левый, а второй — правый,
- вершины без потомков называются *листьями*, остальные вершины называются *внутренними вершинами*.

## 3 Игра для произвольной функции

Обобщим игру Алисы и Боба на случай произвольной функции от двух аргументов. Пусть $X$, $Y$ и $Z$ — непустые конечные множества. Цель Алисы и Боба в игре для функции $f\colon X \times Y \to Z$ заключается в вычислении значения $f$ на входах $x \in X$ и $y \in Y$.

**Задача 3.1** (1 балл). *Как вычислить $f : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}^k$ за $2n$ сообщений?*

**Задача 3.2** (1 балл). *Как вычислить $f : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}^k$ за $n + k$ сообщений?*

**Задача 3.3** (2 балла). *Пусть $f : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}^n$ и $f(x,y) = x$. Докажите, что всегда будет пара $(x,y)$, на которой Алиса и Боб пошлют не менее $n$ сообщений.*

В последней задаче по сути требуется доказать, что не существует более эффективного способа решить эту задачу, чем способ, при котором Алиса посылает Бобу весь $x$ целиком. Для доказательства более сложных утверждения такого рода, нам нужно формально определить, что значит, что Алиса и Боб договорились о некотором способе решить задачу. Будем говорить, что Алиса и Боб договорились, если они определили *протокол общения*.

**Определение 3.1 (Протокол общения)**

Пусть $X$, $Y$ и $Z$ — это три произвольных непустых конечных множества, а $f\colon X \times Y \to Z$ — некоторая функция. *Протокол общения* $\Pi$ для функции $f$ — это упорядоченное корневое двоичное дерево со следующими пометками:

- каждая внутренняя вершина помечена буквой «А» или «Б»,
- каждое ребро к левому потомку помечено нулём, к правому — единицей,
- каждый лист помечен элементом множества $Z$.

Для каждой внутренней вершины $v$ с пометкой «А» определена функция $A_v\colon X \to \{0,1\}$, а для каждой внутренней вершины $u$ с пометкой «Б» определена функция $B_u\colon Y \to \{0,1\}$.
*Результат* протокола $\Pi$ на входе $(x,y)$ обозначается $\Pi(x,y)$, и определяется, как пометка конечной вершины пути $\pi(x,y)$, построенного по следующим правилам:

- первая вершина пути $\pi(x,y)$ — это корень,
- каждая следующая вершина пути является потомком предыдущей, причём
  - каждая вершина пути $v$ с пометкой «А» соединена с потомком ребром с пометкой $A_v(x)$
  - каждая вершина $u$ с пометкой «Б» соединена с потомком ребром с пометкой $B_u(y)$
- последняя вершина пути $\pi(x,y)$ — лист.

Протокол $\Pi$ называется *корректным* протоколом для функции $f$, если для каждой пары входов $(x,y)$ выполняется $\Pi(x,y) = f(x,y)$.

Протокол описывает общение игроков на всех возможных входах. Пометки во внутренних вершинах — это указание на игрока, который посылает сообщение, пометки на рёбрах — это посылаемые сообщения, пометки в листьях — это результат вычисления $f(x,y)$, а функции в вершинах — это правила, по которым игроки выбирают сообщение, которое нужно послать в данный момент. Каждой паре входов $(x,y)$ соответствует путь

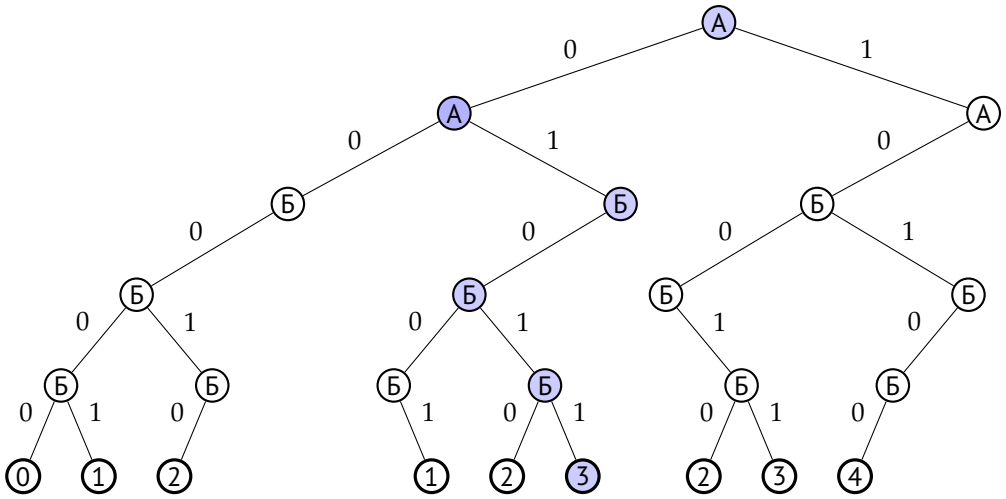



Рис. 3. Дерево протокола для решения задачи 1.0 описанного в разборе. Синим цветом выделены вершины пути $\pi(1,2)$.

$\pi(x,y)$ от корня к некоторому листу, который задаётся описанными выше правилами. Если протокол корректный, то для каждой пары входов $(x,y)$ путь $\pi(x,y)$ заканчивается в листе с пометкой $f(x,y)$.

**Задача 3.4** (2 балла). *Пусть* $\Pi$ *— некоторый протокол для функции* $f\colon X \times Y \to Z$*, и пусть на входах* $(x_1,y_1)$ *и* $(x_2,y_2)$ *пути* $\pi(x_1,y_1)$ *и* $\pi(x_2,y_2)$ *заканчиваются в одном и том же листе. Докажите, что* $\pi(x_1,y_2)$ *и* $\pi(x_2,y_1)$ *заканчиваются в том же листе.*

**Задача 3.5** (1 балл). *Докажите, что если Ева, подслушивающая общение Алисы и Боба, знает протокол общения, то она может восстановить* $f(x,y)$ *не зная* $x$ *и* $y$*.*

**Определение 3.2**

*Сложностью функции* $f$ называется наименьшая глубина протокола, вычисляющего функцию $f$ (обозначается $C(f)$). (Глубина дерева — это максимальная рёберная длина пути от корня до листа.)

Функция $\mathrm{EQ}_n\colon \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$ проверяет две битовые строки длины $n$ на равенство: $\mathrm{EQ}_n(x,y) = 1$ тогда и только тогда, когда $x = y$.

**Задача 3.6** (4 балла). *Докажите, что* $C(\mathrm{EQ}_n) = n+1$.
*(В этой задаче нужно показать, что любой протокол для* $\mathrm{EQ}_n$ *будет глубины не меньше* $n+1$*. Попробуйте оценить минимальное число листьев в таком протоколе.)*

Функция РАЗНОСТЬ$_n\colon \{0,\dots,2^n-1\} \times \{0,\dots,2^n-1\} \to \{-2^n+1,\dots,2^n-1\}$ вычисляет разность целых чисел $x$ и $y$: РАЗНОСТЬ$_n(x,y) = x - y$.

**Задача 3.7** (4 балла). *Докажите, что* $C(\text{РАЗНОСТЬ}_n) = 2n$.

**Задача 3.8** (5 баллов). *Функция* $\mathrm{DISJ}_n\colon \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$ *проверяет, есть ли позиция, в которой и у Алисы, и у Боба стоят единицы:* $\mathrm{DISJ}_n(x,y) = 0$ *тогда и только тогда, когда существует* $i \in \{1,\dots,n\}$ *такое, что* $x[i] = y[i] = 1$ *(здесь и далее* $x[i]$ *обозначает* $i$*-й бит строки* $x$*). Докажите, что существует такая константа* $c > 0$*, что* $C(\mathrm{DISJ}_n) \geq c \cdot n$*.*

Функция СРЕДНИЙ$_n(x,y)\colon \{0,1\}^n \times \{0,1\}^n \to \{1,\dots,n\}$, определяет центральный элемент в упорядоченной по возрастанию последовательности, содержащей все номера позиций, на которых в строках $x$ и $y$ встречаются единицы (если в позиции $i$ у обоих игроков

стоят единицы, то $i$ в последовательности встречается дважды). Если число элементов нечётно и равно $2m+1$, функция возвращает элемент на позиции $m+1$. В случае, если число элементов чётно и равно $2m$, функция возвращает элемент на позиции $m+1$.

**Задача 3.9** (3 балла)**.** *Докажите, что существует такая константа $c > 0$, что для любого $n = 2^k$ выполняется $C(\text{СРЕДНИЙ}_n) \leq c \cdot k^2$.*

**Задача 3.10** (10 баллов)**.** *Докажите, что существует такая константа $c > 0$, что для любого $n = 2^k$ выполняется $C(\text{СРЕДНИЙ}_n) \leq c \cdot k$.*

# 4 Коммуникационная сложность

Область теоретической информатики, которая занимается такими задачами про Алису и Боба, называется *коммуникационная сложность*. Она была разработана Эндрю Яо в 1979 году для доказательства сложности некоторых задач параллельных вычислений, а в дальнейшем нашла применения и других областях компьютерных наук. Коммуникационная сложность естественным образом возникает в потоковых и распределённых алгоритмах, схемной сложности и сложности доказательств, и в ряде других областей. Основная литература по коммуникационной сложности написана на английском. На русском можно прочесть небольшой обзор области тут, а также главу про коммуникационную сложность в этой книге.

## 4.1 Базовые утверждения и понятия

Мы уже ввели определение коммуникационного протокола 3.1 и коммуникационной сложности 3.2. Нам также понадобятся другие понятия для нашего удобства.

Входное пространство коммуникационной задачи можно вопринимать как матрицу. Каждой функции $f$ будем сопоставлять матрицу $X \times Y$, в которой в клетке $(x_i, y_j)$ стоит значение $f(x_i, y_j)$.

**Утверждение 4.1**

Рассмотрим дерево протокола со входом из множества $X \times Y$. Рассмотрим в нём произвольную вершину $u$. Тогда все входы, из которых можно прийти в вершину $u$, образуют прямоугольник $R_u = X_u \times Y_u \subseteq X \times Y$.

*Доказательство.* Это можно доказать двумя способами.

*Первый способ:* пусть на входах $(x_1, y_1)$ и $(x_2, y_2)$ мы приходим в вершину $u$. Тогда нетрудно убедиться, что на входе $(x_1, y_2)$ Алиса и Боб будут делать те же действия, что и на входах $(x_1, y_1)$ и $(x_2, y_2)$ соответственно. Отсюда видно, что входы, приводящие в вершину $u$, образуют прямоугольник $R_u = X_u \times Y_u \subseteq X \times Y$.

*Второй способ:* Рассмотрим таблицу элементов $X \times Y$. После первого хода Боба табличка делится пополам горизонтальной линией, так как при одних $x \in X$ Боб посылает Алисе 1, а при других — 0. Потом Алиса посылает свой бит Бобу, и каждый из двух получившихся прямоугольников делится своей вертикальной прямой, и так далее. В итоге мы получим разбиение $X \times Y$ на непересекающиеся прямоугольники, и каждый из этих прямоугольников соответствует листу в коммуникационном протоколе. □

Про прямоугольник $R_u$ можно думать в следующим образом: если мы находимся в вершине протокола $u$, то нам необходимо решить задачу (то есть построить протокол) для всех входов из прямоугольника $R_u$. В частности этот подход можно рассмотреть, как комбинаторное определение протокола: бинарное дерево, в котором каждой вершине сопоставлен прямоугольник входов. И если вершины $a, b$ являются потомками $u$, то $R_u \subseteq R_a \cup R_b$.

**Определение 4.1**

Прямоугольник $R \subset X \times Y$ называется одноцветным для отношения $F$, если существует $z \in Z$, что для всех $(x, y) \in R$ верно $(x, y, z) \in F$. Такой прямоугольник будем называть $z$-одноцветным.

Рассмотрим величину $\chi_0(f)$, равную минимальному числу прямоугольников, которыми можно дизъюнктно покрыть нули в матрице. Аналогично определяется $\chi_1(f)$. Тогда

листьев в коммуникационном протоколе будет хотя бы $\chi(f) = \chi_0(f) + \chi_1(f)$. Эти рассуждения дают следующую оценку:

$$C(f) \geq \log \chi(f) = \log(\chi_0(f) + \chi_1(f)).$$

Эта оценка не всегда точна. Это нам даёт понять следующий пример:

*Пример* 1. Рассмотрим такой пример разбиения таблицы $X \times Y$ на прямоугольники: в центре находится прямоугольник из 1, а вокруг него расположены 4 прямоугольника из 0. Покажем, что для этого разбиения не существует дерева протокола. Действительно, рассмотрим первое действие игроков. После него таблица должна поделиться на две части, но на рисунке 4 видно, что нет разреза, проходящего через всю таблицу.

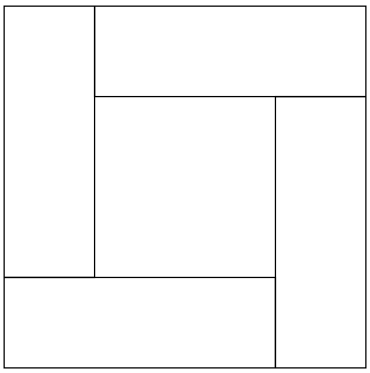
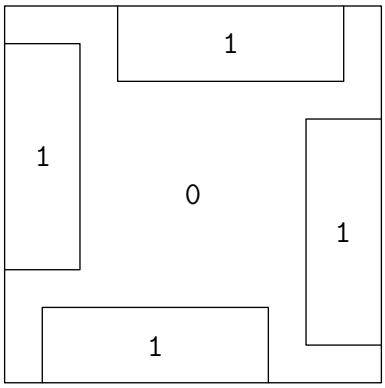


Рис. 4. Разбиение матрицы на одноцветные прямоугольники, которое не соотвествует никакому протоколу.

**Задача 4.1.** *Приведите пример, когда* $L(f) > \chi(f)$.




**Метод трудного множества.** Возьмем некоторый набор входов $(x_1, y_1), \ldots, (x_m, y_m)$, для которого выполнено, что никакие два входа не могут лежать в одном одноцветном прямоугольнике. Тогда для каждой такой пары входов должен быть свой одноцветный прямоугольник в разбиение. Значит, $\chi(f) \geq m$, а следовательно $D(f) \geq \log m$.

**Метод полуаддитивной меры.** Данный метод является обобщение метода трудного множеста. Определим некоторую полуаддитивную меру $\mu$ на подмножествах $X \times Y$ ($\mu(R_1 \cup R_2) \leq \mu(R_1) + \mu(R_2)$). Пусть для любого одноцветного прямоугольника $R$ верно $\mu(R) \leq \alpha$, $\alpha > 0$.

**Утверждение 4.2**

Верно следующее $C(f) \geq \log \dfrac{\mu(X \times Y)}{\alpha}$.

**Метод ранга.** Пусть $M_f$ — матрица некоторой функции $f$ со значениями из $\{0,1\}$. Пусть $x_1, \ldots, x_k$ — листы коммуникационного протокола для функции $f$, а $R_{x_1}, \ldots, R_{x_n}$ — соответствующие им прямоугольники в $M_f$. Из алгебры мы знаем, что $\operatorname{rank}(M_f) \leq \sum \operatorname{rank}(R_i)$, а $\operatorname{rank}(R_i) = 1$ для любого $i$. Отсюда можно сделать вывод, что количество листьев в протоколе не меньше $\operatorname{rank} M_f$, а коммуникационная сложность — не меньше $\log \operatorname{rank}(M_f)$.

## 4.2 Игра для отношения

Обобщим игру Алисы и Боба на случай *трёхместного отношения*. Пусть $X$, $Y$ и $Z$ — непустые конечные множества, а $R \subset X \times Y \times Z$ — трёхместное отношение, в котором для любых $x \in X$ и $y \in Y$ всегда найдётся $z \in Z$ (не обязательно единственный) такой, что $(x,y,z) \in R$. В игре для отношения $R$ цель Алисы и Боба заключается в том, что бы по входам $x \in X$ и $y \in Y$ узнать элемент $z \in Z$ (один и тот же для обоих игроков) такой, что $(x,y,z) \in R$. (Проверьте, что определения 3.1 и 3.2 легко обобщаются на случай такого трёхместного отношения.)

Частным случаем такого трёхместное отношения является *игра Карчмера — Вигдерсона* для функции $f : \{0,1\}^n \to \{0,1\}$, обозначается $\mathrm{KW}_f$ и определяется так:

**Определение 4.2**

*Игра Карчмера — Вигдерсона для функции* $f : \{0,1\}^n \to \{0,1\}$ — это следующая коммуникационная игра: Алиса получает $x \in f^{-1}(0)$, Боб получает $y \in f^{-1}(1)$, и они вместе пытаются найти такое $i \in [n]$, что $x_i \neq y_i$. Иначе говоря, игра Карчмера — Вигдерсона — это коммуникационная задача для *отношения*

$$\mathrm{KW}_f = \{((x,y),i) \mid x \in f^{-1}(0), y \in f^{-1}(1), x_i \neq y_i\}.$$

Отношение $\mathrm{KW}_f$ называется *отношением Карчмера — Вигдерсона* для функции $f$.

Функция чётности $\oplus_n : \{0,1\}^n \to \{0,1\}$ определяется так:

$$\oplus_n(x) = x[1] + x[2] + \cdots + x[n] \mod 2.$$

Другими словами, что $\oplus_n(x) = 0$ тогда и только тогда, когда в $x$ чётное число единиц. В игре для отношения $\mathrm{KW}_{\oplus_n}$ Алиса получает строку с чётным числом единиц, а Боб — с нечётным. Их задача найти такое число $i$, что соответствующие биты $x[i]$ и $y[i]$ различны. Заметим, что такое $i$ всегда существует.

**Задача 4.2.** *Для $n = 2^k$ покажите, что $C(\mathrm{KW}_{\oplus_n}) \leq 2k$.*

Функция логического «или» $\vee_n : \{0,1\}^n \to \{0,1\}$ определяется так:

$$\vee_n(x) = x[1] \vee x[2] \vee \cdots \vee x[n].$$

Другими словами, $\vee_n(x) = 0$ только для строки состоящей из всех нулей. На всех остальных входах $\vee_n$ принимает значение 1.

**Задача 4.3.** *Для $n = 2^k$ покажите, что $C(\mathrm{KW}_{\vee_n}) = k$.*

## 4.3 Формулы и коммуникационная сложность

В этой главе мы посмотрим на применение коммуникационной сложности для доказательства оценок на формульную сложность.

**Определение 4.3**

*Формула в базисе Де Моргана для функции* $f\colon \{0,1\}^n \to \{0,1\}$ — это булевая формула с переменными $\{x_1, \ldots, x_n\}$, соответствующим отдельным битам входа $f$, и со связками (гейтами) $\{\wedge, \vee, \neg\}$, вычисляющая функцию $f$. Законы Де Моргана позволяют нам предполагать, что все $\neg$ находятся непосредственно перед переменными. Структура формулы Де Моргана представляет собой корневое дерево (листья соответствуют переменным, а внутренние вершины — логическим связкам). Размером формулы называется количество листьев, а глубиной формулы — высота дерева, т.е. количество рёбер в самом длинном простом пути от корня до некоторого листа.

**Определение 4.4**

Будем говорить, что семейство булевых функций $f_n : \{0,1\}^n \to \{0,1\}$ вычисляется формулами Де Моргана размера $s(n)$, если для каждого $n \in \mathbb{N}$ существует формула Де Моргана размера $s(n)$, вычисляющая $f_n$. Формульной сложностью $L(f)$ функции $f$ называется минимальная функция $s$, такая что $f$ вычисляется формулами Де Моргана размера $s(n)$.

**Определение 4.5**

Будем говорить, что семейство булевых функций $f_n : \{0,1\}^n \to \{0,1\}$ вычисляется формулами Де Моргана глубины $d(n)$, если для каждого $n \in \mathbb{N}$ существует формула Де Моргана глубины $d(n)$, вычисляющая $f_n$. Формульной глубиной $D(f)$ функции $f$ называется минимальная функция $d$, такая что $f$ вычисляется формулами Де Моргана глубины $d(n)$.

Есть некоторая связь между этими двумя характеристиками.

**Утверждение 4.3**

Для любой булевой функции $f$ верно

$$\log_2 L(f) \leq D(f) \leq 3 \log_2 L(f).$$

Оказывается, что формульная сложность функции $f$ связана с коммуникационной сложностью для отношения $\mathrm{KW}_f$. Мы формулируем теорему Карчмера — Вигдерсона, которая связывает две эти сложности.

**Теорема 4.1 (Карчмер — Вигдерсон)**

Для каждой формулы $\phi$ вычисляющей $f$, существует такой протокол $\Pi_\phi$ для отношения Карчмера — Вигдерсона $\mathrm{KW}_f$, что его дерево совпадает с деревом, описывающим структуру формулы $\phi$. Верно и обратное: если есть протокол для $\mathrm{KW}_f$, то есть и формула для $f$ с такой же структурой.

## 4.4 Задачи для разминки

**Задача 4.4.** *У Алисы имеется $n$-битная строка $x$, а у Боба $n$-битная строка $y$. Известно, что $y$ получен из $x$ инвертированием одного бита.*

*а) Придумайте детерминированный коммуникационный протокол сложности $\mathcal{O}(\log n)$, который позволяет Бобу узнать $x$.*

b) *Придумайте однораундовый детерминированный коммуникационный протокол сложности $\mathcal{O}(\log n)$, который позволяет Бобу узнать $x$. (В однораундовом протоколе Алиса посылает некоторое сообщение Бобу, после чего Боб вычисляет результат).*

**Задача 4.5.** *Пусть дан граф $G$ без петель. Алиса и Боб получают две вершины данного графа $x, y$ и хотят узнать существует ли ребро $(x, y)$. Докажите, что детерминированная сложность данной задачи не менее $\log \chi(G)$, где $\chi(G)$ — хроматическое число графа $G$.*
Подсказка: *попробуйте предъявить хорошую раскраску, если есть короткий коммуникационный протокол.*

**Задача 4.6.** *Докажите, что $C(\mathrm{CIS}_G) = \mathcal{O}(\log^2 n)$. Где $x$ интерпретируется как характеристическая функция некоторой клики в графе $G$, а $y$ — как характеристическая функция некоторого независимого множества в графе $G$. $\mathrm{CIS}_G(x, y) = 1$, если клика и независимое множество имеют общую вершину, обе стороны знают граф $G$.*

**Задача 4.7.** *Постройте детерминированный коммуникационный протокол, который вычисляет функцию GT, передавая в среднем константу битов. Функция $\mathrm{GT}(x, y)$ определена на парах $x, y$ целых чисел в интервале $\{0, \ldots, 2^n - 1\}$ и принимает значение 1, если $x > y$, и значение 0, иначе. Говоря о среднем, мы имеем в виду, что $x, y$ выбираются случайно и независимо среди всех чисел указанного интервала с равномерным распределением.*

**Определение 4.6**

*Внутреннее произведение* $\mathrm{IP}_n \colon \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$ задаётся соотношением

$$\mathrm{IP}_n(x, y) = x[1]y[1] + x[2]y[2] + \cdots + x[n]y[n] \mod 2.$$

**Задача 4.8.** *Докажите, что коммуникационная сложность* IP *равна $n - \mathcal{O}(1)$.*

## 4.5 Исследовательские задачи

Задачи в таких секциях являются исследовательскими. Их решение может потребовать значительно больше времени и сил. Предложенные задачи имеют различную сложность. В частности, решение открытых задач неизвестно. Поэтому перед тем, как браться за эти задачи, стоит подумать над решением обычных задач в секциях 4.5, 5.6, **??**.

Функция подсчёта $\mathrm{MOD}p_n : \{0,1\}^n \to \{0,1\}$ для натурального $p > 1$ определяется следующим соотношением $\mathrm{MOD}p_n(x) = 0 \iff x[1] + x[2] + \cdots + x[n] = 0 \mod p$. Отметим, что $\oplus_n$ — это в точности $\mathrm{MOD}2_n$.

**Открытая задача 4.9 (очень сложно)**

Предлагается улучшить верхнюю оценку из статьи Andrew Chin для отношения $\mathrm{KW}_{\mathrm{MOD}p_n}$ для конкретного значения $p > 2$.

a) Для $p = 3$ лучше $2.881 \log_2 n$,

b) Для $p = 5$ лучше $3.475 \log_2 n$,

c) Для $p = 11$ лучше $4.930 \log_2 n$.

**Задача 4.10.** *Улучшите константу в балансировке протоколов. Доказать, что $C(f) \leq c \log_2 L(f)$ для $c < 3$.*

# 5 Коммуникационная сложность с оракулом

В этой главе мы будем исследовать коммуникационную сложность с оракулом. В данной модели Алиса и Боб отправляют сообщения третьему игроку Чарли, выполняющему роль оракула, он вычисляет некоторую функцию $g$ и отправляет результат игрокам, цель Алисы и Боба — вычислить с помощью Чарли функцию $f$.

Одним из наиболее исследованных оракулов в коммуникационной сложности является задача равенства EQ. Она является сложной для детерминированной коммуникационной сложности, но вероятностно решается за константное количество раундов в модели с публичными случайными битами. В коммуникационной сложности с оракулом EQ Алиса и Боб каждый раунд отправляют некоторые битовые строки третьему игроку, Чарли, который сообщает в ответ, равны ли они (см. рис. 5). Будем обозначать за $C^{\mathrm{EQ}}(f)$ коммуникационную сложность функции $f$ с оракулом EQ, т.е. минимальную глубину протокола, который вычисляет функцию $f$ с оракулом EQ. Впервые данная модель была введена [BFS86].

**Задача 5.1.** *Покажите, что $C(f) \geq C^{\mathrm{EQ}}(f)$. Примером лучшего разделения может служить сама же задача* EQ*, $C^{\mathrm{EQ}}(\mathrm{EQ}) = 1$, $C(\mathrm{EQ}) = \Theta(n)$.*

Для подробного изучения модели с оракулом можно почитать статью Chattopadhyay и других. Также ознакомиться со всеми доказательствами из этой главы можно тут.

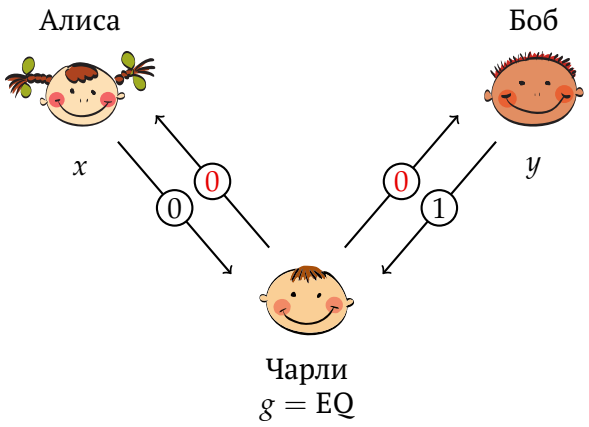



Рис. 5. Общение Алисы и Боба с оракулом Чарли.

**Задача 5.2.** *Придумайте решение задачи 1.0 за 2 раунда в модели с оракулом* EQ*.*

**Разбор задачи 5.2.** Алиса и Боб отправляют Чарли первый бит двоичной записи $x$ и $y$. Чарли говорит им, равны ли они, таким образом каждый из игроков понимает какой бит был у другого. Игроки проделывают такой же раунд со втором битом своих чисел и узнают числа друг друга.

## 5.1 Протокол с оракулом

Пусть $A$ — семейство коммуникационных задач $A_m : \{0,1\}^m \times \{0,1\}^m \to \{0,1\}$ для $m \in \mathbb{N}$. Если входы игроков $(x,y) \in \{0,1\}^n \times \{0,1\}^n$, то каждое сообщение в модели с оракулом $A$ — это пара входов $(g_1(x), g_2(y)) \in \{0,1\}^m \times \{0,1\}^m$ для функции $A_m$, где $g_1$ и $g_2$ выбраны заранее, а выход $A_m(g_1(x), g_2(y))$ виден обоим игрокам. Сложность такого

протокола — это число вызовов оракула. $C^A(f)$ — это минимальная сложность по всем протоколам для функции $f$.

Фактически, в нашем анализе после обращения к оракулу мы разбиваем набор входов на набор прямоугольников. Для доказательства нижних оценок нам будет удобнее работать с более сильной моделью, в которой все возможные наборы ответов заранее разбиты на прямоугольники, и оракул сообщает игрокам не только ответ на их запрос, но и к какому прямоугольнику в разбиении относятся их вход, без каких-либо дополнительных затрат.

Заметим, что вызов функции $A_m$ со входом, преобразованным $g_1$ и $g_2$, эквивалентен вызову функции $B = A_m \circ (g_1, g_2)$, и что матрица $B$ может быть получена из матрицы $A_m$ путем удаления, дублирования и перестановки некоторых строк или столбцов. Каждой матрице $M$ оракула мы сопоставляем некоторое разбиение $\mathcal{R}(M)$ матрицы на одноцветные прямоугольники. В общем, таких вариантов может быть много; правильный выбор будет иметь решающее значение для нашей техники доказательства нижней оценки. Единственное требование заключается в том, что это разбиение на одноцветные прямоугольники.

Протокол в модели с оракулом $A$, вычисляющий функцию $f : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$ — это дерево, где каждая вершина соответствует прямоугольнику $R \subseteq \{0,1\}^n \times \{0,1\}^n$ входов. Каждая вершина связана с матрицей $M$ оракула тех же размеров, что и $R$, и имеет по одному сыну для каждого прямоугольника $R' \in \mathcal{R}(M)$. Находясь в вершине, помеченной $R$, игроки переходят к сыну с прямоугольником $R'$, который содержит их входы. Каждый лист помечен 0 или 1, а метка листа $R$ равна $f(x,y)$ для каждого $(x,y) \in R$.

Аналогично тому, как один бит детерминированной коммуникации обновляет разделение входного пространства, где каждый прямоугольник делится на два. Один вызов оракула обновляет разделение пространства, где каждый прямоугольник $R$ заменяется на разбиение $\mathcal{R}(M(R))$, связанное с матрицей оракула $M(R)$ того же размера. Это значит, что все начинается с одного прямоугольника $\mathcal{R}_0 = \{\{0,1\}^n \times \{0,1\}^n\}$, и после обращения к оракулу, получается раздел $\mathcal{R}_i = \cup_{R \in \mathcal{R}_{i-1}} \mathcal{R}(M(R))$. Если протокол вычисляет функцию $f$ после $C$ вызовов, то разбиение $\mathcal{R}_C$ является разбиением матрицы $M_f$ функции $f$ на одноцветные прямоугольники.

Для простоты изложения предположим, что мы ограничиваем возможный вход оракула $A$ длиной не более $n$ (т.е. у игроков есть доступ к оракулам $A_m$, где $m \leq n$, или что тоже самое $g_1, g_2 : \{0,1\}^n \to \{0,1\}^m$, $m \leq n$). Но все рассуждения можно обобщить на случай неограниченного $m$.

## 5.2 Оракул единичного расстояния Хэмминга

Следующие секции посвящены изучению сложности функций в модели с оракулом расстояния Хэмминга.

**Определение 5.1**

Точное расстояние Хэмминга $\mathrm{EHD}_k(x,y) = 1$ тогда и только тогда, когда расстояние Хэмминга (количество различающихся битов) между $x$ и $y$ равно ровно $k$, где $x, y \in \{0,1\}^n$.

Для разбиения $\mathcal{R} = \bigcup R_i$, где $R_i = A_i \times B_i$, обозначим за $p(\mathcal{R}) = \sum_{R_i} |A_i| + |B_i|$ полупериметр разбиения $\mathcal{R}$. За $p(M)$ обозначим минимальный периметр по всем разбиениям матрицы $M$ на одноцветные прямоугольники.

**Лемма 5.1**

Для матрицы $M$ оракула EQ размера $a \times b$ существует разбиение $\mathcal{R}$ на одноцветные прямоугольники такое, что $p(\mathcal{R}) \leq 2(a+b)\log(a+b)$.

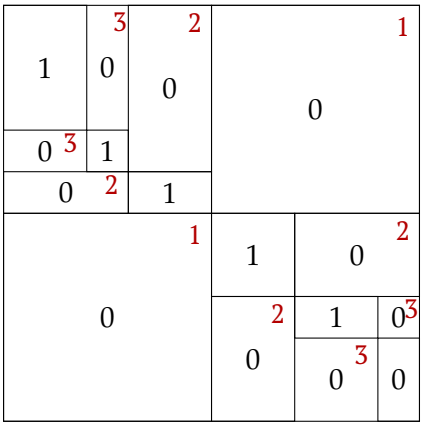


Рис. 6. Матрица оракула EQ.

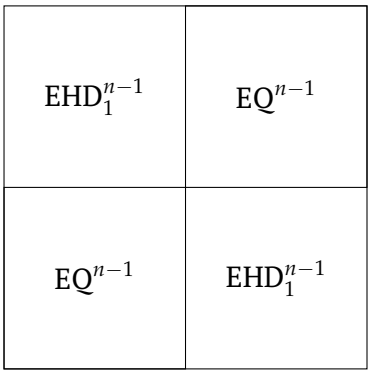



Рис. 7. Матрица $\mathrm{EHD}_1$.

**Лемма 5.2**

Пусть $f : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$ — булева функция, которая в модели с оракулом $\mathrm{EHD}_1$ имеет сложность $C$. Тогда существует разбиение $\mathcal{R}$ коммуникационной матрицы $f$ на одноцветные прямоугольники с периметром $p(\mathcal{R}) \leq 2^{n+1}(2n^2)^C$.

Теперь нам нужна оценка на размер одноцветных прямоугольников матрицы $\mathrm{EHD}_k$, в случае константного $k$ мы докажем лемму 5.3. Будем доказывать двойной индукцией по параметру функции $k$ и размеру входа $n$.

**Лемма 5.3**

Для любого 1-прямоугольника $R$ матрицы $\mathrm{EHD}_k$ верно $|R| \leq 2n^k$.

**Теорема 5.1**

Коммуникационная сложность $C^{\mathrm{EHD}_1}(\mathrm{EHD}_k)$ не менее $\frac{k}{5}$.

## 5.3 Оракул точного расстояния Хэмминга равного $\ell$

**Лемма 5.4**

Пусть $f : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$ — булева функция, которая в модели с оракулом $\mathrm{EHD}_\ell$ имеет сложность $C$. Тогда существует разбиение $\mathcal{R}$ коммуникационной матрицы $f$ на одноцветные прямоугольники с периметром $p(\mathcal{R}) \leq 2^{n+1}(2n^{\ell+1})^C$.

**Теорема 5.2**

Коммуникационная сложность $C^{\mathrm{EHD}_\ell}(\mathrm{EHD}_k)$ не менее $\frac{k}{2(\ell+2)}$.

## 5.4 Верхняя оценка с оракулом расстояние Хэмминга не более $\ell$

**Определение 5.2**

$\mathrm{HD}_{\leq k}(x,y) = 1$ тогда и только тогда, когда расстояние Хэмминга между $x$ и $y$ не более $k$.

Несложно заметить, что матрица задачи $\mathrm{HD}_{\leq k}$ имеет схожую $\mathrm{EHD}_k$ структуру, поэтому полученная оценка верна и для задачи $\mathrm{HD}_{\leq k}$ с оракулом $\mathrm{HD}_{\leq \ell}$.

Несложно понять, что используя оракул $\mathrm{HD}_{\leq \ell}$ как EQ задача $\mathrm{HD}_{\leq k}$ может быть решена за $2k \cdot \log n$. Аналогичное верно и для задачи $\mathrm{EHD}_k$ с оракулом $\mathrm{EHD}_\ell$. Непонятно как использовать оракул $\mathrm{EHD}_\ell$ более эффективно. Для случая задачи $\mathrm{HD}_{\leq k}$ с оракулом $\mathrm{HD}_{\leq \ell}$ получается доказать более точную верхнюю оценку.

**Теорема 5.3**

$$\mathrm{P}^{\mathrm{HD}_{\leq \ell}}(\mathrm{HD}_{\leq k}) \leq 2 \cdot \frac{k}{\ell} \cdot \log \ell \cdot \log n.$$

## 5.5 Оракул однобитового равенства

Для получения нижних оценок на формулы в полном булевом базисе можно переносить нижние оценки на коммуникационную сложность игр Карчмера — Вигдерсона в модели с однобитовым оракулом $\mathrm{EQ}_1$.

По формуле в полном булевом базисе для функции $f$ можно получить протокол для $KW_f$ с оракулом $\mathrm{EQ}_1$ такой же глубины. Алиса получает $x \in f^{-1}(0)$, Боб $y \in f^{-1}(1)$, если формула $\phi = \phi_1 \wedge \phi_2$ ($\phi = \phi_1 \vee \phi_2$) Алиса (Боб) отправляет в какой подформуле у нее (него) $\phi_i(x) = 0$ ($\phi_i(y) = 1$) и они переходят в нужную подформулу. Если $\phi = \phi_1 \oplus \phi_2$, то Алиса отправляет 1 в оракул $\mathrm{EQ}_1$, если $\phi_1(x) = 1$ и $\phi_2(x) = 1$, и 0, если $\phi_1(x) = 0$ и $\phi_2(x) = 0$, Боб отправляет 1 в оракул, если $\phi_1(y) = 0$ и $\phi_2(y) = 1$, и 0, иначе. Тогда если биты Алисы и Боба равны, то они идут в левую подформулу $\phi_1$, иначе идут в правую подформулу $\phi_2$. Таким образом, получаем протокол для $\mathrm{KW}_f$ в модели с оракулом $\mathrm{EQ}_1$ такой же глубины как и формула для $f$, значит $C^{\mathrm{EQ}_1}(\mathrm{KW}_f) \leq C(f)$.

Рассмотрим сложность булевых функций от $2n$ бит с разделенным входом для Алисы и Боба. Покажем, что сложность случайной функции равняется $n - o(n)$.

**Теорема 5.4**

Существует функция $f : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$, которая имеет сложность $C^{\mathrm{EQ}_1}(f) = n - o(n)$.

**Задача 5.3.** $C^{\mathrm{EQ}_1}(\mathrm{EHD}_1) = n/2 + \mathcal{O}(1)$.

## 5.6 Исследовательские задачи

**Задача 5.4.** *Оцените сложность $C^{\mathrm{EQ}_1}(\mathrm{EHD}_k)$ в зависимости от $k$.*

**Открытая задача 5.5 (сложно)**

Оцените сложность $C^{\mathrm{EQ}}(\mathrm{EHD}_k)$ в зависимости от $k$.

**Открытая задача 5.6 (сложно)**

Попробуйте улучшить нижнюю оценку на $C^{\mathrm{EHD}_\ell}(\mathrm{EHD}_k)$.

**Открытая задача 5.7**

Попробуйте улучшить верхнюю оценку на $C^{\mathrm{EHD}_\ell}(\mathrm{EHD}_k)$ и $C^{\mathrm{HD}_{\leq \ell}}(\mathrm{HD}_{\leq k})$.

## Список литературы

[BFS86] Laszlo Babai, Peter Frankl, and Janos Simon. Complexity classes in communication complexity theory. In *27th Annual Symposium on Foundations of Computer Science (sfcs 1986)*, pages 337–347, 1986.