

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ  
ФЕДЕРАЦИИ

Федеральное государственное автономное образовательное учреждение  
высшего образования

УРАЛЬСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ  
имени первого Президента России Б. Н. Ельцина

ИНСТИТУТ ЕСТЕСТВЕННЫХ НАУК И МАТЕМАТИКИ

Кафедра алгебры и фундаментальной информатики

**Различение слов перестановочными автоматами**

Направление подготовки 02.04.02

«Фундаментальная информатика и информационные технологии»

Допустить к защите:

Зав. кафедрой:

доктор физико-математических  
наук

профессор М.В. Волков

---

Нормоконтролер:

кандидат физико-математических  
наук

доцент И.И. Иванов

---

Магистерская диссертация

**Карповой**

**Ольги Дмитриевны**

---

Научный руководитель:

доктор физико-математических  
наук,

профессор А.М. Шур

---

Екатеринбург

2018

# Содержание

<b>ВВЕДЕНИЕ</b>	<b>2</b>
<b>1 Основные определения и постановка задачи</b>	<b>3</b>
1.1 Основные определения . . . . .	3
1.2 Постановка задачи . . . . .	4
1.3 Сведение к поиску тождеств в полугруппах и группах . . . .	5
1.4 Полезные факты . . . . .	7
<b>2 Серии тождеств</b>	<b>8</b>
2.1 Тождества из двух блоков . . . . .	9
2.2 Тождества из трех блоков . . . . .	10
2.3 Тождества из четырех и более блоков . . . . .	24
<b>3 Заключение</b>	<b>25</b>

# ВВЕДЕНИЕ

Задача о различении двух наборов входных данных - одна из самых простых вычислительных задач, которые можно себе представить. Обычно, входные данные представлены в виде двух строк  $u$  и  $v$  над конечным алфавитом  $\Sigma$  и известны заранее.

В мощной вычислительной модели, такой как RAM, задача решается за константную память (помимо памяти, занимаемой строками): нам достаточно одного регистра для того, чтобы найти позицию, в которой различаются  $u$  и  $v$ .

Однако для более слабой модели, например, для конечных автоматов, задача существенно усложняется, и ее уже нельзя решить за константную память. Задача об определении минимального размера конечного автомата, различающего два данных слова, NP-трудна. Более того, не всё известно об асимптотике минимального количества вершин, необходимого, чтобы построить различающий автомат для каждой пары слов, длина которых меньше либо равна заданному наперед числу. Это и есть задача о различении слов автоматами, и на данный момент известны только верхняя и нижняя границы искомой функции, между которыми довольно большой разрыв.

Далее в работе будут даны все необходимые определения, задача будет поставлена более формально, также будет показана эквивалентность этой задачи поиску тождеств в полгруппе преобразований и группе перестановок. Более того, будут приведены серии тождеств, одна из которых позволяет слегка поднять нижнюю границу искомой функции.

# 1 Основные определения и постановка задачи

## 1.1 Основные определения

Пусть  $\mathcal{A} = (\Sigma, Q, \delta, s, T)$  – детерминированный конечный автомат, где  $\Sigma$  – входной алфавит, из которого формируются слова, принимаемые автоматом,  $\Sigma \neq \emptyset$ ,

$Q$  – множество состояний автомата,  $Q \neq \emptyset$ ,

$\delta$  – функция переходов, определенная как отображение  $\delta : Q \times \Sigma \rightarrow Q$ ,

$s$  – начальное состояние,  $s \in Q$

$T$  – множество терминальных (конечных) состояний,  $T \subseteq Q$ .

**Определение 1.1.** Автомат принимает слово, если по окончании его обработки он находится в терминальном состоянии.

**Определение 1.2.** Пусть  $u$  и  $v$  – слова над алфавитом  $\Sigma$ . Говорят, что автомат различает слова  $u$  и  $v$ , если он принимает одно из них и не принимает другое.

**Пример 1.1.** Автомат на рисунке 1.1 различает слова 0010 и 1000.

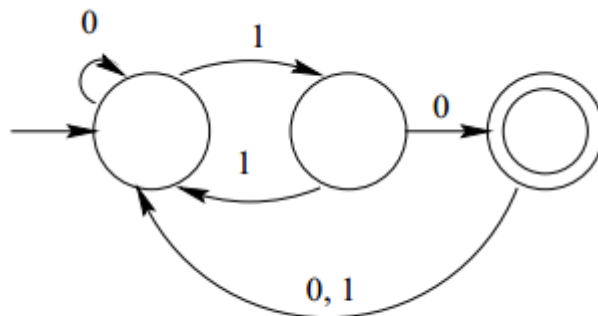


Рис. 1:

**Определение 1.3.** Пару слов  $(u, v)$  будем называть тождеством для некоторого автомата, если он либо принимает и  $u$ , и  $v$ , либо отвергает и  $u$ , и  $v$ , то есть не различает их.

Однако, в дальнейшем нам будет удобнее будет считать терминальными все состояния у рассматриваемых автоматов, и в связи с этим подкорректировать определения.

**Определение 1.4.** Автомат различает слова  $u$  и  $v$ , если он заканчивает их чтение в разных состояниях, то есть  $s.u \neq s.v$ .

Поскольку вся работа посвящена перестановочным автоматам, необходимо дать и это ключевое определение.

**Определение 1.5.** Автомат  $\mathcal{A}$  – перестановочный, если переход из любого состояния по любому символу является перестановкой состояний, или, что тоже самое, для любого символа  $x$  из  $\Sigma$  и любых состояний  $q$  и  $p$   $\delta(q, x) \neq \delta(p, x)$ .

## 1.2 Постановка задачи

Обозначим за  $sep(u, v)$  количество состояний в минимальном детерминированном конечном автомате, различающем  $u$  и  $v$ , за  $sep_p(u, v)$  – количество состояний в перестановочном автомате, различающем  $u$  и  $v$ .

**Пример 1.2.** Можно проверить, что ни один автомат из двух состояний не сможет различить слова 0010 и 1000, а как мы видели из рисунка 1.1, существует автомат с тремя состояниями, различающий эту пару. Значит,  $sep(0010, 1000) = 3$ .

Обозначим

$$S(n) = \max_{u \neq v; |u|, |v| \leq n} sep(u, v)$$

и

$$S_p(n) = \max_{u \neq v; |u|, |v| \leq n} sep_p(u, v)$$

Задача о различении слов автоматами, известная как Separating Words Problem, состоит в том, чтобы найти хорошую асимптотическую оценку

функций  $S(n)$  и  $S_p(n)$ , то есть оценить, сколько состояний должно быть в автомате, различающем две строки длины  $n$ .

Эту задачу сформулировали Павел Горальчик и Вацлав Коубек в 1986 году [1, 2]. Они же доказали, что  $S(n) = o(n)$ . Позже этой задачей занимался Джон Робсон, который в 1989 году [3] доказал, что  $S(n) = O(n^{2/5}(\log n)^{3/5})$  для произвольных автоматов, и в 1996 году [4] опубликовал статью, в которой было доказано, что  $S_p(n) = O(n^{1/2})$ .

### 1.3 Сведение к поиску тождеств в полугруппах и группах

Обозначим за  $T_k$  полугруппу всех отображений множества  $1, \dots, k$  в самого себя относительно операции композиции отображений.  $T_k$  называют полугруппой преобразований на  $k$  элементах.

**Определение 1.6.** Тождеством в полугруппе  $T$  называют пару слов  $(u, v)$  такую, что образы  $u$  и  $v$  под действием любого отображения  $\Sigma \rightarrow T$  совпадают как элементы  $T$ . Длина тождества  $(u, v)$  - максимум длин  $u$  и  $v$ . Факт тождественности  $u$  и  $v$  в  $T_k$  будем обозначать как  $u \equiv_k v$ .

**Определение 1.7.** Полугруппа преобразований детерминированного конечного автомата  $\mathcal{A}$  - это подполугруппа  $T_{|Q|}$ , состоящая из всех отображений  $w : q \rightarrow q.w$ , где  $q \in \Sigma^*$ .

Следующий факт соединяет между собой тождества в полугруппах и разделение слов автоматами.

**Факт 1.1.** Для любой пары слов  $u, v$ ,  $u \equiv_k v \iff S(u, v) > k$

Действительно, если  $u \equiv_k v$ , то это тождество выполняется и для полугруппы преобразований любого конечного детерминированного автомата с

$k$  состояниями, откуда получаем  $q.u = q.v$  для любого состояния  $q$ . С другой стороны, если для некоторого отображения  $\rho : \Sigma \rightarrow T_k$   $\rho(u) \neq \rho(v)$  в  $T_k$ , тогда преобразования  $\rho(a), a \in \Sigma$  могут быть использованы для создания автомата с  $k$  состояниями, отличающего  $u$  от  $v$ .

Известно, что задача о проверке  $u \equiv_k v$  принадлежит классу coNP-complete для любого  $k > 2$ . Поэтому исходя из Факта 1.1, задача о проверке  $S(u, v) \leq k$  является NP-полной. К тому же, задача о различении слов автоматами эквивалентна поиску асимптотики минимальной длины тождества в  $T_k$ .

До недавнего времени самым коротким тождеством в  $T_k$  было

$$x^{k-1} = x^{k-1+lcm(k)}, \quad (1)$$

где  $lcm(k)$  - это наименьшее общее кратное всех чисел от 1 по  $k$ . Однако недавно было найдено новое тождество, которое короче 1, если  $k$  - простое или степень простого числа [5].

Имеет место аналогичная Факту 1.1 связь различения слов перестановочными автоматами и тождеств в группе перестановок  $S_k$ .

**Факт 1.2.** Для любой пары слов  $u, v$ ,  $u \equiv_k v$  в  $S_k \iff S_p(u, v) > k$

Самое короткое тождество в  $S_k$ , дающее почву для нижней оценки функции  $S_p(k)$  имеет вид

$$x^{lcm(k)} = 1$$

Было несколько попыток найти более короткие групповые тождества. Существование тождества длины  $O(e^{\sqrt{n \log n}})$  было доказано в статье [6]; основная идея заключается в применении функции Ландау о максимальном порядке перестановки. Совсем недавно было также доказано существование тождества длины  $O(e^{\log^4 n \log \log n})$  [7], основанного на новых результатах о диаметре графа Кэли  $S_k$ . В данной работе будет представлена серия тождеств, показывающая, что  $S_p(n) \geq \frac{3}{2} \log n + o(\log n)$ .

## 1.4 Полезные факты

Вставить сюда ограничения: одинаковая длина слов, бинарный алфавит, начинаются и заканчиваются по-разному над алфавитом (ху) (ух)



## 2 Серии тождеств

Известно, что группа перестановок  $S_k$  удовлетворяет тождеству 1 и его бинарному собрату  $x^{lcm(k)} \equiv_k y^{lcm(k)}$ .

## 2.1 Тождетсва из двух блоков

## 2.2 Тождества из трех блоков

$(xy)^a(yx)^b(xy)^c = (yx)^c(xy)^b(yx)^a$  является тождеством в  $S_k$ , если для каждого  $z$  - порядка  $k$ -перестановки выполняется хотя бы одно из следующих правил:

$$z|a \text{ и } z|(b - c) \quad (2)$$

$$z|c \text{ и } z|(b - a) \quad (3)$$

$$z|b \text{ и } z|(a + c) \quad (4)$$

**Определение 2.1.** Циклической перестановкой из  $k$  элементов с шагом  $s$  будем называть такую циклическую перестановку, в которой элемент с номером  $i$  переходит в элемент с номером  $i + s \pmod k$ .

Далее будем считать, что элементы перестановки длины  $k$  - это числа от 0 до  $k - 1$ .

**Лемма 2.1.** Пусть  $(xy)^a(yx)^b(xy)^c = (yx)^c(xy)^b(yx)^a$  - тождество в  $S_k$ , где  $k$  - нечетное. Тогда  $a - b + c \equiv 0 \pmod k$ .

*Доказательство.* Зафиксируем перестановки  $xy$  и  $yx$  такие, что  $xy$  - циклическая перестановка с шагом -1, а  $yx$  - циклическая перестановка с шагом 1. Рассмотрим перестановочный автомат, в котором переход по символам осуществляется соответствующими перестановками  $x$  и  $y$ . Тогда, чтобы  $(xy)^a(yx)^b(xy)^c = (yx)^c(xy)^b(yx)^a$  было тождеством для такого автомата, требуется, чтобы автомат закончил читать обе части равенства в одном состоянии, то есть

$$(-a) + b + (-c) \equiv c + (-b) + a \pmod k \quad (5)$$

что эквивалентно

$$2(a - b + c) \equiv 0 \pmod k \quad (6)$$

Из того, что  $k$  нечетно, следует

$$(a - b + c) \equiv 0 \pmod{k}$$

. □

**Лемма 2.2.** Пусть  $(xy)^a(yx)^b(xy)^c = (yx)^c(xy)^b(yx)^a$  - тождество в  $S_k$ , где  $k$  - четное. Тогда  $a - b + c \equiv 0 \pmod{\frac{k}{2}}$ .

*Доказательство.* Рассмотрим перестановочный автомат, как в доказательстве Леммы 2.1. Аналогично, получим

$$2(a - b + c) \equiv 0 \pmod{k} \tag{7}$$

Из того, что  $k$  четно, следует

$$(a - b + c) \equiv 0 \pmod{\frac{k}{2}}$$

. □

**Следствие 2.1.** Пусть  $(xy)^a(yx)^b(xy)^c = (yx)^c(xy)^b(yx)^a$  - тождество в  $S_k$ . Тогда  $a - b + c \equiv 0 \pmod{\frac{\text{lcm}(k)}{2}}$ .

*Доказательство.* Из Леммы 2.1  $a - b + c \equiv 0$  по модулю наименьшего общего кратного всех нечетных чисел, меньших  $k$ . По Лемме 2.2  $a - b + c \equiv 0$  по модулю предмаксимальной степени числа 2, не превосходящей  $k$ . Отсюда следует, что  $a - b + c \equiv 0 \pmod{\frac{\text{lcm}(k)}{2}}$ . □

**Теорема 2.1.** Пусть  $(xy)^a(yx)^b(xy)^c = (yx)^c(xy)^b(yx)^a$  - тождество в  $S_k$  и  $a + b + c \leq \frac{\text{lcm}(k)}{2}$ . Тогда  $b = a + c$ .

*Доказательство.* Сразу заметим, что ни одно из чисел  $a, b, c$  не равно 0, так как в противном случае мы будем рассматривать тождество другого типа.

Из Следствия 2.1 вытекает, что

$$b - a - c = \frac{\text{lcm}(k)}{2}m,$$

где  $m \in \mathbb{Z}$ . Откуда

$$b = a + c + \frac{\text{lcm}(k)}{2}m > 0.$$

Получим цепочку неравенств

$$0 < a + c + \frac{\text{lcm}(k)}{2}m < \frac{\text{lcm}(k)}{2} + \frac{\text{lcm}(k)}{2}m = \frac{\text{lcm}(k)}{2}(m + 1),$$

то есть  $m > -1$ .

С другой стороны, подставим  $b$  в  $a + b + c$ , получим

$$2(a + c) + \frac{\text{lcm}(k)}{2}m \leq \frac{\text{lcm}(k)}{2}$$

или

$$2(a + c) \leq \frac{\text{lcm}(k)}{2}(1 - m)$$

Поскольку сумма  $a$  и  $c$  должна быть положительным числом, требуется

$$\frac{\text{lcm}(k)}{2}(1 - m) > 0,$$

то есть  $m < 1$ .

Значит, при заданных ограничениях  $m = 0$ , что влечет  $b = a + c$ .  $\square$

**Следствие 2.2.** Для кратчайшего тождества вида  $(xy)^a(yx)^b(xy)^c = (yx)^c(xy)^b$  выполняется  $b = a + c$ .

*Доказательство.* Достаточно показать, что существуют тождества, для которых  $a + b + c \leq \frac{\text{lcm}(k)}{4}$  (тогда кратчайшее тождество также удовлетворяет этому условию, а по теореме для всех тождеств с таким свойством выполняется  $b = a + c$ ).

Пусть  $m = 2k/3$ ,  $a := \text{lcm}(m)$ ,  $c := \text{lcm}(k - m) \cdot P(m)$ ,  $b := a + c$ , где  $P(m)$  - произведение всех простых и степеней простых чисел из множества  $\{m + 1, \dots, k\}$ .  $a$  и  $c$  взяты из доказательства длины тождества из двух блоков. Оттуда же понятно, что любой порядок перестановки делит или  $a$ , или  $c$ , а, благодаря выбору  $b$ , делит и соответствующую разность. Длина

такого тождества, конечно, в два раза больше длины тождества из двух блоков  $(e^{\frac{2}{3}k+O(\frac{k}{\log k})})$ , однако все равно асимптотически меньше, чем  $\frac{\text{lcm}(k)}{4}$  (который равен  $e^{k+O(\frac{k}{\log k})}$ ).  $\square$

Теперь, вооружившись утверждением о связи показателей степеней рассматриваемого тождества, можно доказать обратное утверждение, т.е. если  $(xy)^a(yx)^b(xy)^c \equiv_k (yx)^c(xy)^b(yx)^a$ , то выполняется хотя бы одно из условий 2 - 4. Для этого нам понадобится доказать еще несколько лемм. Однако заметим сразу, что равенство  $b = a + c$  делает истинной вторые части утверждений 2 - 4, если первые истинны.

**Лемма 2.3.** *Если  $(xy)^a(yx)^b(xy)^c \equiv_k (yx)^c(xy)^b(yx)^a$ , тогда хотя бы одно из чисел  $a, b, c$  делится на 2.*

*Доказательство.* От противного. Пусть  $a, b, c$  нечетны. Тогда  $b \neq a + c$ , поскольку их четность не совпадает. Противоречие.  $\square$

**Лемма 2.4.** *Если  $(xy)^a(yx)^b(xy)^c \equiv_k (yx)^c(xy)^b(yx)^a$  и  $k \geq 4$ , тогда хотя бы одно из чисел  $a, b, c$  делится на 3.*

*Доказательство.* От противного. Пусть ни одно из чисел  $a, b, c$  не делится на 3. Тогда возможны лишь два варианта:

1.  $a \equiv_3 c \equiv_3 1$  и  $b \equiv_3 2$

2.  $a \equiv_3 c \equiv_3 2$  и  $b \equiv_3 1$

(В остальных случаях хотя бы одно из чисел оказывается кратным трем)

Рассмотрим автомат относительно символов  $xy, yx$  на рисунке 2. Такой автомат можно получить, взяв за перестановку по  $x$   $(0)(1, 2, 3)$ , по  $y$  -  $(1)(3, 2, 0)$ . Начальное состояние 0.

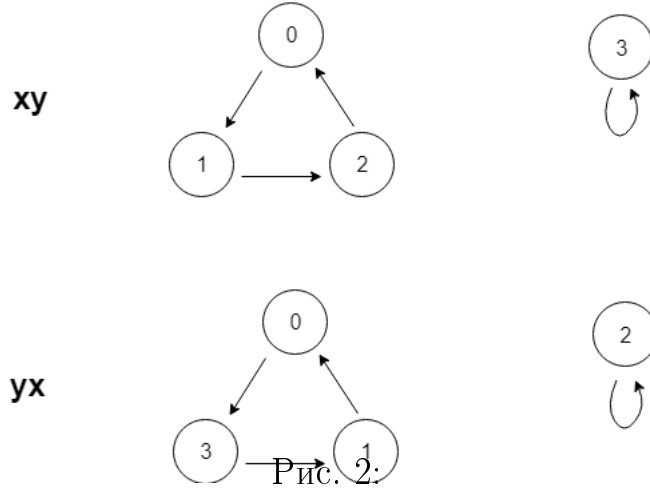


Рис. 2.

В первом случае автомат закончит читать левую часть тождества в состоянии 3 (после прочтения  $(xy)^a$  окажется в состоянии 1, затем, прочитав  $(yx)^b$  придет в состояние 3 в цикле), а правую - в состоянии 1 (после прочтения  $(yx)^c$  окажется в состоянии 3 и в нем останется после  $(xy)^b$ ).

Во втором случае автомат закончит читать левую часть тождества в состоянии 1 (после прочтения  $(xy)^a$  окажется в состоянии 2, затем, прочитав  $(yx)^b$  останется в состоянии 2), а правую - в состоянии 2 (после прочтения  $(yx)^c$  окажется в состоянии 1 и перейдет в состояние 2 после  $(xy)^b$ ).

Получили противоречие с тем, что данная пара - тождество.  $\square$

**Лемма 2.5.** Если  $(xy)^a(yx)^b(xy)^c \equiv_k (yx)^c(xy)^b(yx)^a$  и  $k \geq 4$ , тогда хотя бы одно из чисел  $a, b, c$  делится на  $k$ .

*Доказательство.* От противного. Пусть ни одно из чисел  $a, b, c$  не делится на  $k$ . Разберем несколько случаев.

1.  $2a \equiv_k 0$ .

Поскольку мы предположили, что  $a \not\equiv_k 0$ , значит  $a \equiv_k \frac{k}{2}$ . Так как  $a + c = b$ ,  $b \not\equiv_k 0$  по предположению, то  $c \not\equiv_k a$ .

Рассмотрим автомат  $\mathcal{A}$ , в котором  $xy$  - перестановка с шагом 1, в которой поменяли местами  $a$  и  $c$ , а  $yx$  - перестановка с шагом -1 (см. Рисунок 3). Данный автомат различит слова  $(xy)^a(yx)^b(xy)^c$  и  $(yx)^c(xy)^b(yx)^a$ .

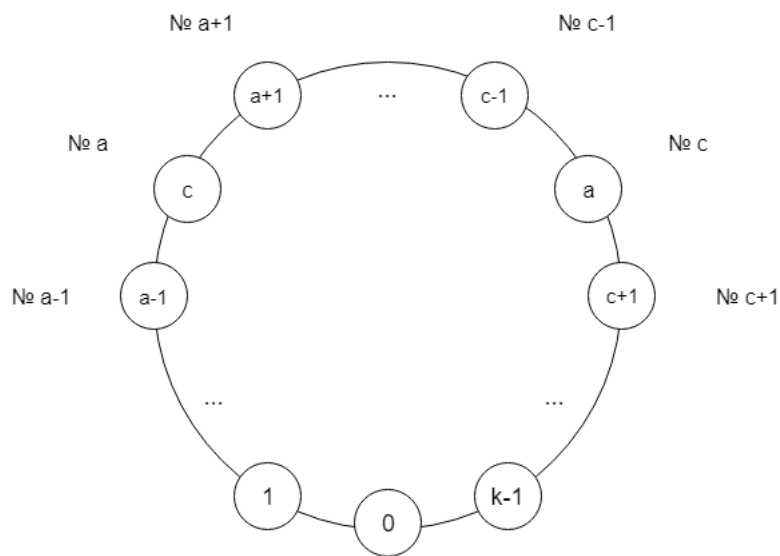


Рис. 3: Цикл  $xy$  автомата  $\mathcal{A}$

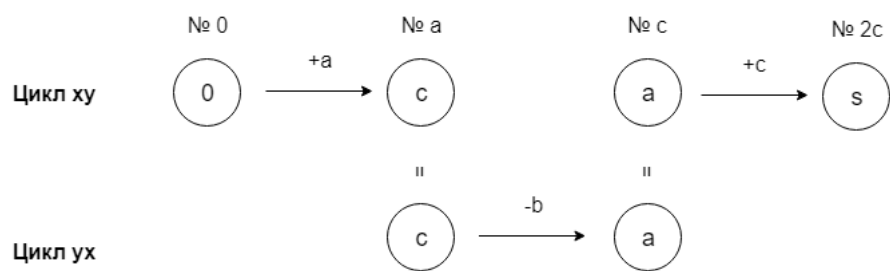


Рис. 4: Чтение слова  $(xy)^a(yx)^b(xy)^c$  автоматом  $\mathcal{A}$

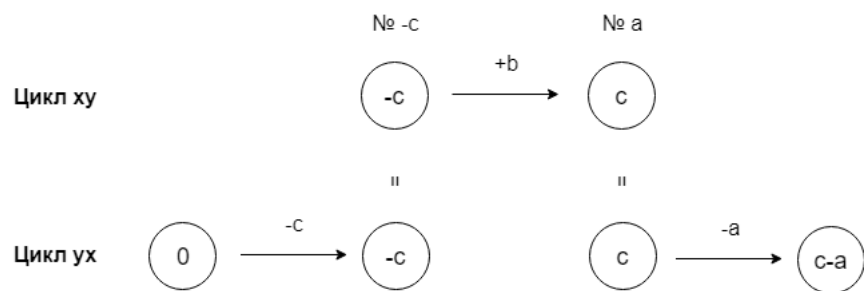


Рис. 5: Чтение слова  $(yx)^c(xy)^b(yx)^a$  автоматом  $\mathcal{A}$



$\mathcal{A}$  закончит читать второе слово в состоянии  $c - a$  (см. Рисунок 5).

Корректность переходов:

- $-c \not\equiv_k a$  и  $-c \not\equiv_k c$  поскольку иначе в обоих случаях он был бы равен  $\frac{k}{2}$  и совпадал бы с  $a$ ;
- $b - c = a$  из Теоремы 2.1.

$\mathcal{A}$  закончит читать второе слово в состоянии с номером  $2c$  (см. Рисунок 5). Корректность переходов:

- $c - b \equiv_k -a$  из Теоремы 2.1, а  $a \equiv_k -a$  поскольку  $a \equiv_k \frac{k}{2}$ ;
- $2c \not\equiv_k c$  поскольку по предположению  $c \not\equiv_k 0$ .

Конечное состояние при чтении  $(yx)^c(xy)^b(yx)^a$  зависит от того, совпадает ли  $2c$  с  $a$  по модулю  $k$ :

- $2c \equiv_k a$ , тогда состоянием с номером  $2c$  будет состояние  $c$ , которое не совпадает с  $c - a$ , поскольку  $-a$  не сравнимо с нулем по модулю  $k$ ;
- $2c \not\equiv_k a$ , тогда конечным состоянием будет  $2c$ , которое не совпадает с  $c - a$ , поскольку  $a + c \not\equiv_k 0$ .

## 2. $2a \not\equiv_k 0$ и $2a \not\equiv_k -c$

Рассмотрим автомат  $\mathcal{B}$ , в котором  $xy$  - перестановка с шагом 1, а  $yx$  - перестановка с шагом -1, в которой поменяли местами  $a$  и  $2a$  (см. Рисунок 6). Данный автомат различит слова  $(xy)^a(yx)^b(xy)^c$  и  $(yx)^c(xy)^b(yx)^a$ .

$\mathcal{B}$  закончит читать  $(xy)^a(yx)^b(xy)^c$  в состоянии  $a$  (см. Рисунок 7). Корректность переходов:

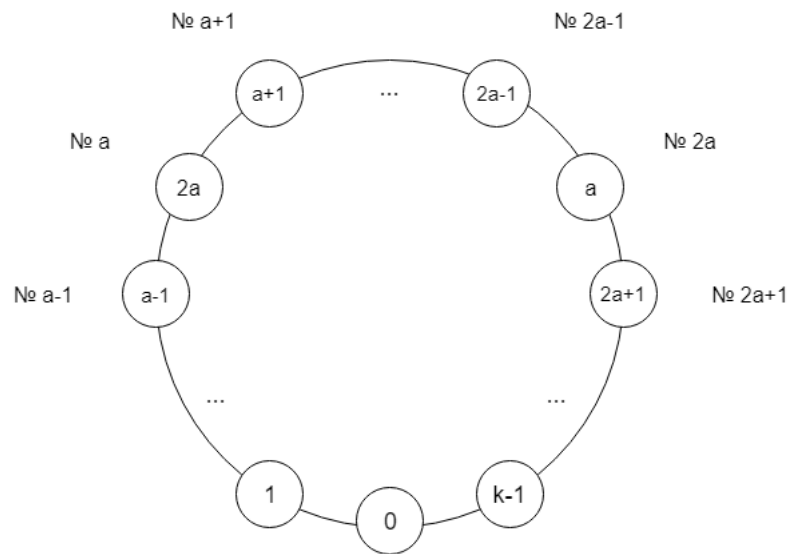


Рис. 6: Цикл  $yx$  автомата  $\mathcal{B}$

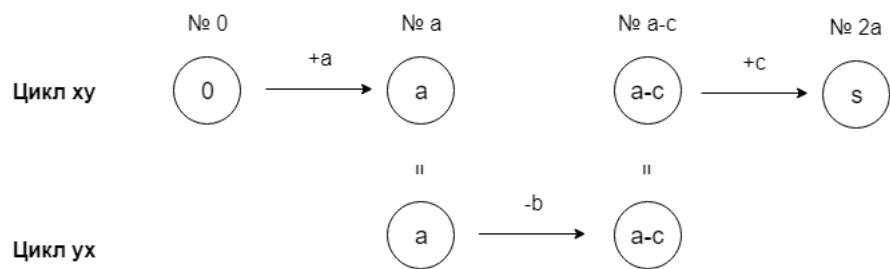


Рис. 7: Чтение слова  $(xy)^a (yx)^b (xy)^c$  автоматом  $\mathcal{B}$

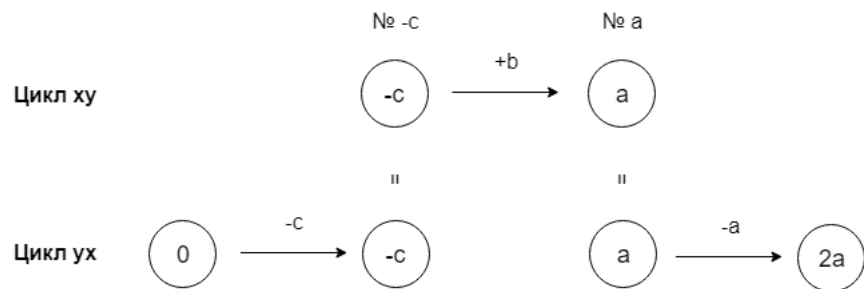


Рис. 8: Чтение слова  $(yx)^c (xy)^b (yx)^a$  автоматом  $\mathcal{B}$

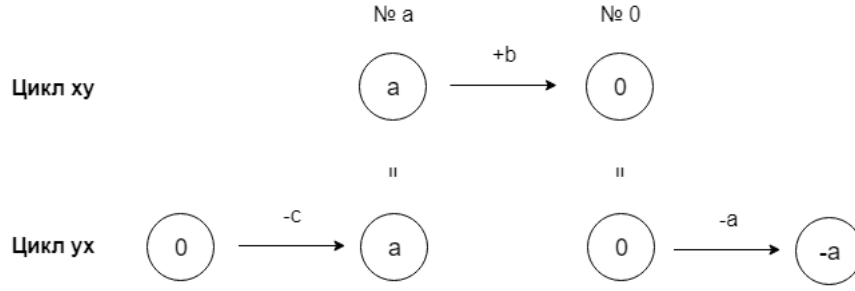


Рис. 9: Чтение слова  $(yx)^c(xy)^b(yx)^a$  автоматом  $\mathcal{B}$  в случае, когда  $2a \equiv_k -c$

- $2a - b = a - c$  по Теореме 2.1,  $a - c \not\equiv_k a$ , так как  $-c \not\equiv_k 0$  по предположению, и  $a - c \not\equiv_k 2a$ , так как  $a + c \not\equiv_k 0$  также по предположению.

$\mathcal{B}$  закончит читать  $(yx)^c(xy)^b(yx)^a$  в состоянии  $2a$  (см. Рисунок 8).

Корректность переходов:

- $-c \not\equiv_k a$ , поскольку  $a + c \not\equiv_k 0$  по предположению, и  $-c \not\equiv_k 2a$  по заданному ограничению;
- $b - a = c$  по Теореме 2.1.

Состояния  $a$  и  $2a$  не совпадают, так как  $a \not\equiv_k 0$  по предположению.

### 3. $2a \not\equiv_k 0$ и $2a \equiv_k -c$ и $3a \not\equiv_k 0$

Рассмотрим автомат  $\mathcal{B}$  из предыдущего случая (см. Рисунок 6). В этом случае этот автомат также разделит рассматриваемую пару слов.

Чтение слова  $(xy)^a(yx)^b(xy)^c$  будет абсолютно таким же, как и в предыдущем случае. Чтение слова  $(yx)^c(xy)^b(yx)^a$  закончится в состоянии  $-a$  (см. Рисунок 9). Корректность переходов:

- $2a \equiv_k -c$  по заданному ограничению;
- $a + b \equiv_k 0$  поскольку  $a + b = 2a + c \equiv_k 2a - 2a = 0$  по Теореме 2.1 и заданному ограничению;
- $-a \not\equiv_k a$ , так как  $2a \not\equiv_k 0$ , и  $-a \not\equiv_k a$ , поскольку  $3a \not\equiv_k 0$  по заданному ограничению.



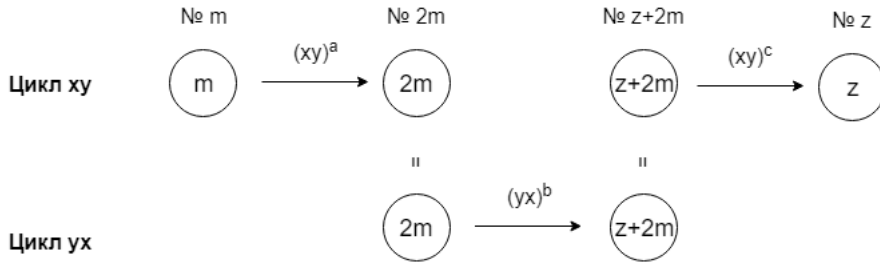


Рис. 11: Чтение слова  $(xy)^a(yx)^b(xy)^c$  автоматом  $\mathcal{C}$  при  $a \equiv_k \frac{k}{3}$

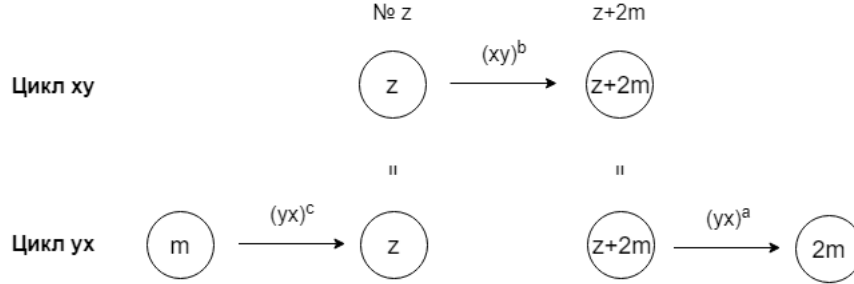


Рис. 12: Чтение слова  $(yx)^c(xy)^b(yx)^a$  автоматом  $\mathcal{C}$  при  $a \equiv_k \frac{k}{3}$

- $z + 2m \not\equiv_k 2m$ , так как  $z \not\equiv_k 0$  по выбору  $z$ .

Состояние  $z$  не совпадает с  $2m$  по выбору  $z$ .

(b)  $a \equiv_k c \equiv_k 2m$ ,  $b \equiv_k m$

Начальным состоянием в автомате  $\mathcal{C}$  назовем состояние 0.

$\mathcal{C}$  закончит читать  $(xy)^a(yx)^b(xy)^c$  в состоянии  $z$  (см. Рисунок 13),

а слово  $(yx)^c(xy)^b(yx)^a$  - в состоянии  $2m$  (см. Рисунок 14).

Корректность переходов:

- $z + m \not\equiv_k z$ , так как  $m \equiv_k 2a \not\equiv_k 0$  по заданному ограничению;
- $z + m \not\equiv_k m$ , так как  $z \not\equiv_k 0$  по выбору  $z$ .

Состояние  $z$  не совпадает с  $2m$  по выбору  $z$ .

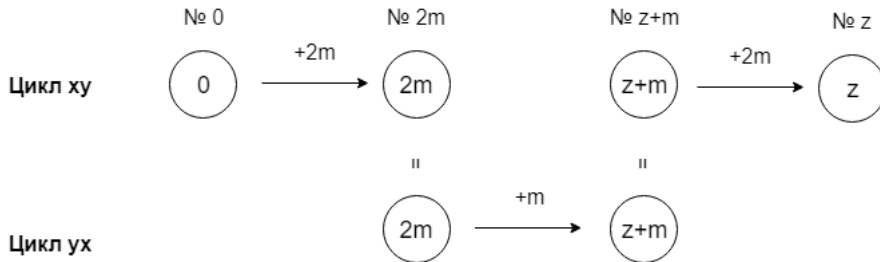


Рис. 13: Чтение слова  $(xy)^a(yx)^b(xy)^c$  автоматом  $\mathcal{C}$  при  $a \equiv_k \frac{2k}{3}$

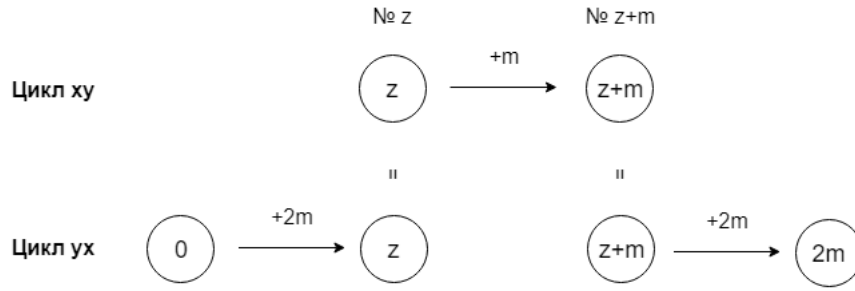


Рис. 14: Чтение слова  $(yx)^a(xy)^b(yx)^a$  автоматом  $\mathcal{C}$  при  $a \equiv_k \frac{2k}{3}$

Рассмотрены все возможные случаи, и для каждого из них приведен автомат, разделяющий пару слов, которая по условию была тождеством. Значит, наше предположение о том, что ни одно из чисел  $a, b, c$  не делится на  $k$ , было ложным.  $\square$

**Теорема 2.2.** Если  $(xy)^a(yx)^b(xy)^c \equiv_k (yx)^c(xy)^b(yx)^a$  и  $k \geq 4$ , то для любого порядка  $ord$  перестановки из  $k$  элементов хотя бы одно из чисел  $a, b, c$  делится на  $ord$ .

*Доказательство.* Если перестановка представима в виде одного цикла (здесь опускаются тривиальные циклы из одного состояния), значит порядок перестановки  $ord$  меньше либо равен  $k$ , и утверждение доказано Леммами 2.3 - 2.5.

Пусть  $k$ -перестановка представима в виде произведения двух циклов с длинами  $c_1, c_2$ . Тогда порядок перестановки равен  $lcm(c_1, c_2)$ . От противного, предположим, что ни одно из чисел  $a, b, c$  не делится на  $lcm(c_1, c_2)$ . Рассмотрим несколько вариантов.

1.  $c_1|a$  и  $c_2|b$ .

Тогда  $c_1 \nmid c$  и  $c_2 \nmid c$ , поскольку иначе разность  $b - c$  делилась бы на  $c_1$  или  $c_2$  соответственно, а это повлекло бы за собой или  $c_2|a$ , или  $c_1|b$ . Также, исходя из Теоремы 2.1,  $c_1|(b - c)$ , откуда  $b \equiv_{c_1} c$ .

Рассмотрим автомат  $\mathcal{A}$  с переходами по  $xy$  и  $yx$ , изображенный на рисунке 15. Начальное состояние автомата - 0; циклы, длинами  $c_1$  и  $c_2$ , у

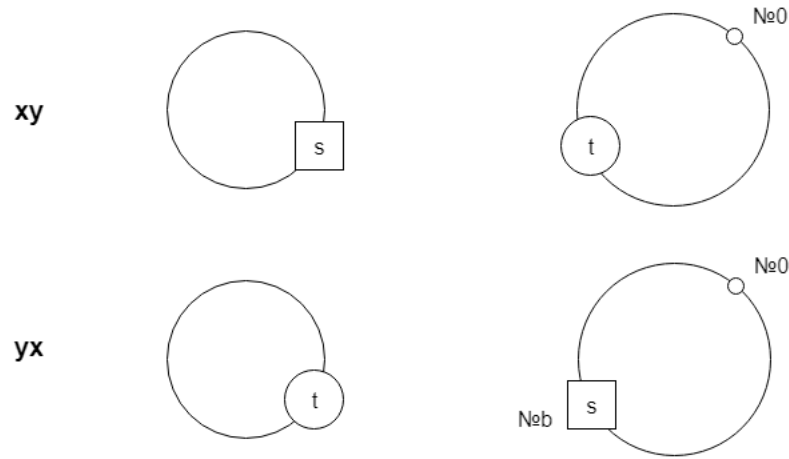


Рис. 15: Автомат  $\mathcal{A}$

$c_1$

разных перестановок отличаются только одним состоянием (состояние  $s$  у перестановки  $xy$  располагается в цикле  $c_2$ , а у перестановки  $yx$  - в цикле  $c_1$ ; аналогичная ситуация с состоянием  $t$ ), остальные вершины у циклов с одинаковыми длинами совпадают. Покажем, что данный автомат различит строки  $(xy)^a(yx)^b(xy)^c$  и  $(yx)^c(xy)^b(yx)^a$ .

- Читаем  $(xy)^a(yx)^b(xy)^c$

$0.(xy)^a = 0$ , так как  $c_1|a$ .  $0.(yx)^b = s$  по выбору вершины  $s$ , и автомат переходит к циклу  $c_2$ .  $s.(xy)^c = w$ , где  $w$  - вершина цикла длины  $c_1$  перестановки  $xy$ , не совпадающая с  $s$ , поскольку  $c_1 \nmid c$ .

- Читаем  $(yx)^c(xy)^b(yx)^a$

$0.(yx)^c = s$ , так как  $b \equiv_{c_1} c$ ; автомат переходит к циклу  $c_2$ .  $s.(xy)^b = s$  поскольку  $c_2|b$ ; автомат переходит к циклу  $c_1$ .  $s.(yx)^a = s$ , поскольку  $c_1|a$ .

Поскольку  $w \neq s$ , автомат различит слова, что противоречит тождественности пары.

2.  $c_1|a$  и  $c_2|$ .

Рассуждая аналогично предыдущему случаю, получим, что  $b \equiv_{c_1} c$ ,  $c_1 \nmid b$ ,  $c_2 \nmid b$ .

Рассмотрим автомат  $\mathcal{A}$  из предыдущего случая, и покажем, что он и здесь различит рассматриваемую пару слов.

- Читаем  $(xy)^a(yx)^b(xy)^c$

$0.(xy)^a = 0$ , так как  $c_1|a$ .  $0.(yx)^b = s$  по выбору вершины  $s$ ; автомат переходит к циклу  $c_2$ .  $s.(xy)^c = s$ , поскольку  $c_2|c$ .

- Читаем  $(yx)^c(xy)^b(yx)^a$

$0.(yx)^c = s$ , так как  $b \equiv_{c_1} c$ ; автомат переходит к циклу  $c_2$ .  $s.(xy)^b = v$ , где  $v$  - вершина цикла длины  $c_2$  перестановки  $xy$ , не совпадающая с  $s$ , поскольку  $c_2 \nmid b$ .  $v.(yx)^a = v$ , поскольку  $c_1|a$ .

Так как  $v$  и  $s$  не совпадают, также приходим к противоречию.

3. Случай  $c_1|c$  и  $c_2|b$  симметричен первому, поэтому можно построить аналогичный автомат. Остальные случаи аналогичны рассмотренным с точностью до смены длин циклов местами.

Получается, что для любой перестановки, представляемой в виде объединения двух циклов, хотя бы одно из чисел  $a, b, c$  делится на наименьшее общее кратное их длин (то есть, делится на обе длины одновременно).

Предположим теперь, что  $k$ -перестановка представима в виде объединения трех циклов с длинами  $c_1, c_2, c_3$ . Для каждой пары циклов, как мы уже доказали, наименьшее общее кратное их длин делит хотя бы одно из чисел  $a, b, c$ . Всего таких пар в этом случае три. В зависимости их распределений по числам  $a, b, c$ , можно также рассмотреть несколько случаев.

1. Все три НОК-а делят один показатель. Тогда этот показатель делится и на  $\text{lcm}(c_1, c_2, c_3)$ ;
2. Два из трех НОК-ов делят один показатель. Без ограничения общности,  $\text{lcm}(c_1, c_2)|a$  и  $\text{lcm}(c_1, c_3)|a$ . Тогда очевидно  $\text{lcm}(c_1, c_2, c_3)|a$ ;



3. На каждый из показателей приходится по одной паре. Без ограничения общности,  $\text{lcm}(c_1, c_2) | a$  и  $\text{lcm}(c_1, c_3) | b$ ,  $\text{lcm}(c_2, c_3) | c$ . Поскольку,  $c_1 | a$  и  $c_1 | b$ ,  $c_1 | (b - a)$ , а по Теореме 2.1  $b - a = c$ , значит и  $c_1 | c$ . Получается, что  $c$  делится на все три длины циклов, а значит делится на их НОК.

Значит, для порядков, соответствующие которым перестановки разбиваются на три цикла, теорема тоже доказана.

Теперь описанные ранее случаи представим как базу индукции по наименьшему числу "циклов"  $n$ , на которые можно разбить порядок (то есть соответствующую ему перестановку). Предположим, что для всех  $n < m$  теорема доказана. Докажем для  $n = m$ . Пусть некоторый порядок есть  $\text{lcm}(c_1, c_2, \dots, c_n)$ . По предположению индукции НОК любых  $n - 1$ -их длин циклов делит хотя бы одно из чисел  $a, b, c$ . Поскольку таких  $n - 1$ -наборов больше чем 3, хотя бы на одно из этих чисел придется хотя бы 2 набора. Без ограничения общности пусть  $\text{lcm}(c_1, c_2, \dots, c_{n-1}) | a$  и  $\text{lcm}(c_2, c_3, \dots, c_n) | a$ . Но тогда  $a$  делится на  $\text{lcm}(c_1, c_2, \dots, c_n)$ . Теорема доказана.  $\square$

## 2.3 Тождества из четырех и более блоков

### 3 Заключение

## Список литературы

- [1] P. Goralčík and V. Koubek. On discerning words by automata // In L. Kott, editor, Proc. 13th Int'l Conf. on Automata, Languages, and Programming (ICALP), volume 226 of Lecture Notes in Computer Science, pages 116–122. Springer-Verlag, 1986.
- [2] E.D. Demaine, S. Eisenstat, J. Shallit, D.A. Wilson. “Remarks on Separating Words // Descriptive Complexity of Formal Systems (DCFS 2011), P. 147-157. LNCS Vol. 6808. Springer, 2011.
- [3] J. M. Robson. Separating strings with small automata // Inform. Process. Lett., 30:209–214, 1989.
- [4] J.M. Robson. Separating words with machines and groups// RAIRO Inform. Théor. App. 30, 81–86 (1996)
- [5] A.A. Bulatov, O. Karpova, A.M. Shur, K. Startsev. (2016). Lower Bounds on Words Separation: Are There Short Identities in Transformation Semigroups?. Electronic Journal of Combinatorics. 24.
- [6] K. Bou-Rabee and D. B. McReynolds. Asymptotic growth and least common multiples in groups. Bull. Lond. Math. Soc., 43(6):1059-1068, 2011.
- [7] G. Kozma and A. Thom. Divisibility and laws in finite simple groups. Mathematische Annalen, 364(1):79-95, 2016.