МИНОБРНАУКИ РОССИИ
ФГБОУ ВО «СГУ ИМЕНИ Н. Г. ЧЕРНЫШЕВСКОГО»

КОМБИНАТОРНАЯ ТЕОРИЯ ПОЛУГРУПП

ЛАБОРАТОРНАЯ РАБОТА

студента 3 курса 331 группы
направления 10.05.01 — Компьютерная безопасность
факультета КНиИТ
Токарева Никиты Сергеевича

Проверил
аспирант ______________ В. Н. Кутин

Саратов 2022

1 Постановка задачи

Цель работы: изучение основных понятий теории полугрупп.

Порядок выполнения работы:

1. Рассмотреть понятия полугруппы, подполугруппы и порождающего множества. Разработать алгоритм построения подполугрупп по по таблице Кэли.
2. Разработать алгоритм построения полугруппы бинарных отношений по заданному порождающему множеству.
3. Рассмотреть понятия подгруппы, порождающего множества и определяющих соотношений. Разработать алгоритм построения полугруппы по порождающему множеству и определяющим соотношениям.

2 Теоретические сведения по рассмотренным темам с их обоснованием

Определение 1. Полугруппа – это алгебра S = (S, ·) с одной ассоциативной бинарной операцией ·, т.е. выполняется

$$(x \cdot y) \cdot z = x \cdot (y \cdot z)$$

для любых $x, y, z \in S$.

Определение 2. Пусть A – произвольное множество, называемое алфавитом. Элементы $a \in A$ называются буквами. Словом над алфавитом A называется конечная последовательность букв $a_1, \ldots, a_n$ алфавита A. Слово без букв называется пустым словом и обозначается символом $\wedge$. Для слова $w = a_1, \ldots, a_n$ число n букв в определяющей его последовательности называется длиной этого слова и обозначается символом $l(w)$.

Обозначим символом A^+ множество всех непустых слов над алфавитом и символом A^* - множество слов $A^* = A^+ \cup \{\wedge\}$. На этих множествах слов определена операция умножения, которая называется операцией **конкатенации** слов и определяется по правилу: любым словам $w_1 = a_1, \ldots, a_n$ и $w_2 = b_1, \ldots, b_m$ операция конкатенации ставит в соответствие слово $w_1 \cdot w_2 = a_1, \ldots a_n b_1, \ldots, b_n$. В результате множество слов А+ с операцией конкатенации образует полугруппу, которая называется **полугруппой слов над алфавитом** A, и множество слов A^* с операцией конкатенации образует полугруппу с единичным элементом $\wedge$, которая называется **моноидом слов над алфавитом** A.

Определение 3. Подмножество X полугруппы S называется подполугруппой, если X устойчиво относительно операции умножения, т.е. $\forall x, y \in X$ выполняется свойство: $x \cdot y \in X$. В этом случае множество X с ограничением на нем операции умножения исходной полугруппы S образует полугруппу.

В силу общего свойства подалгебр пересечение любого семейства X_i $(i \in I)$ подполугрупп полугруппы S является подполугруппой S и, значит, множество $Sub(S)$ всех подполугрупп полугруппы S является системой замыканий. множество X. Такая полугруппа обозначается символом $\langle X \rangle$ и называется подполугруппой S, порождённой множеством X. При этом множество X называется также **порождающим множеством** подполугруппы $\langle X \rangle$. В частности, если $\langle X \rangle = S$, то X называется порождающим множеством полугруппы S и говорят, что множество X порождает полугруппу S.

Для любой конечной полугруппы S найдется такой конечный алфавит A, что для некоторого отображения $\phi : A \to S$ выполняется равенство $\langle \phi(A) \rangle = S$ и, значит, $S \cong A^+/ker\phi$ этом случае множество A называется множеством порождающих символов полугруппы S (относительно отображения $\phi : A \to S$). Если при этом для слов $w_1, w_2 \in A$ выполняется равенство $\phi(w_1) = \phi(w_2)$, т.е. $w_1 \equiv w_2(ker\phi)$, то говорят, что на S выполняется соотношение $w_1 = w_2$ (относительно отображения $\phi : A \to S$).

Очевидно, что в общем случае множество таких соотношений $w_1 = w_2$ для всех пар $(w_1, w_2) \in ker\phi$ будет бесконечным и не представляется возможности эффективно описать полугруппу S в виде полугруппы классов конгруэнции $ker\phi$. Однако в некоторых случаях можно выбрать такое сравнительно простое подмножество $\rho \subset ker\phi$, которое однозначно определяет конгруэнцию $ker\phi$ как наименьшую конгруэнцию полугруппы A^+ , содержащую отношение ρ, т.е. $ker\phi = f_{con}(\rho) = f_{eq}(f_{reg}(\rho))$.

Так как в случае $(w_1, w_2) \in \rho$ по-прежнему выполняется равенство $\phi(w_1) = \phi(w_2)$, то будем писать $w_1 = w_2$ и называть такие выражения **определяющими соотношениями**. Из таких соотношений конгруэнция $ker\phi$ строится с помощью применения следующих процедур к словам $u, v \in A^+$:

1. слово v непосредственно выводится из слова u, если v получается из u заменой некоторого подслова w_1 на слово w_2, удовлетворяющее определяющему соотношению $w_1 = w_2$, т.е. $(u, v) = (xw_1y, xw_2y)$ для некоторых $x, y \in A^*$;
2. слово v выводится из слова u, если v получается из u с помощью конечного числа применения процедуры 1.

Если все выполняющиеся на S соотношения выводятся из определяющих соотношений совокупности ρ, то конгруэнция $ker\phi$ полностью определяется отношением ρ и выражение $< A : w_1 = w_2 : (w_1, w_2) \in \rho >$ называется **копредставлением полугруппы** S.

3 Результаты работы

3.1 Алгоритм 1 – Построение подполугруппы по заданному порождающему множеству

Вход: Полугруппа S с таблицей Кэли $A = (a_{ij})$ размерности $n \times n$ и подмножество $X \subset S$.

Выход: Подполугруппа $\langle X \rangle \subset S$.

Шаг 1. Положим $i = 0$, $X_0 = X$.

Шаг 2. Для X_i вычислим $\overline{X}_l = \{x \cdot y : x \in X_i \wedge y \in X\}$ и положим $X_{i+1} = X_i \cup \overline{X}_l$, где выражение $x \cdot y = a_{xy}$ в таблице Кэли A.

Шаг 3. Вычисляем

$$\langle X \rangle = \bigcup_{i=0}^{\infty} X_i.$$

Оценка сложности алгоритма $O(n^3)$.

3.2 Алгоритм 2 – Построение полугруппы бинарных отношений по заданному порождающему множеству

Вход: Конечное n-элементное множество X бинарных отношений, заданное булевыми матрицами размерности $m \times m$.

Выход: Полугруппа $\langle X \rangle$.

Шаг 1. Необходимо инициализировать список $matrices = []$. Известно, что каждому элементу $x_i \in X$ $(0 \leq i < n)$ соответствует матрица $A_i \in M$, где M – множество матриц A_i $(0 \leq i < n)$, тогда элементы списка $matrices$ будут заданы следующим образом: $matrices[i] = A_i$ $(0 \leq i < n)$. Стоит отметить, что список $matrices$ есть полугруппа $\langle X \rangle$, т.е. в этот список будут добавлятся новые элементы полугруппы.

Шаг 2. Инициализировать список $matrices_{new} = []$.

Шаг 3. Далее возьмем матрицу $A_0 = A_i \in matrices$ и умножим ее на матрицы $B_i \in matrices$ $(0 \leq i < n)$. Таким образом получаем матрицу $C = A_0 \odot B_i$, где $\odot$ – операция поэлементного умножения. Затем добавим матрицу C в список $matrices_{new}$. Необходимо таким образом пройти по всем элементам A_i $(0 \leq i < n)$, получая на каждой итерации матрицу C.

Шаг 4. Инициализировать список $matrices_{check} = matrices$ (т.е. делается копия списка $matrices$). Далее добавляем элементы $el \in matrices_{new}$ в список

$matrices$, если их еще нет в списке $matrices$.
Шаг 5. Если после после шага 4 переменная $matrices = matrices_{check}$, то завершить алгоритм, иначе вернуться к шагу 2.

Оценка сложности алгоритма примерно равна $O((n-1) \cdot n^2 \cdot n) = O((n-1) \cdot n^3)$, так как трудно оценить из-за наличия бесконеного цикла в реализации данного алгоритма.

3.3 Алгоритм 3 – Построение полугруппы по порождающему множеству и определяющим соотношениям

Вход: Конечное множество символов A мощности n и конечное множество R определяющих соотношений мощности m.

Выход: Полугруппа $\langle A|R \rangle$.

Шаг 1. Необходимо инициализировать список $semigroup = []$, в который будут добавлены все элементы $a \in A$.
Шаг 2. Инициализировать список $elements_{new} = []$.
Шаг 3. Далее возьмем элемент $x \in semigroup$ и «умножим» его на $y \in semigroup$, получая новое слово $z = xy$. Далее полученное слово необходимо обработать, используя соотношение $r \in R$. После всех преобразований получим слово $z^{'}$. Добавим полученное слово в список $elements_{new}$.
Шаг 4. Инициализировать список $semigroup_{check} = semigroup$ (т.е. делается копия списка $semigroup$). Далее добавляем элементы $z^{'} \in elements_{new}$ в список $semigroup$, если их еще нет в списке $semigroup$.
Шаг 5. Если после после шага 4 переменная $semigroup = semigroup_{check}$, то завершить алгоритм, иначе вернуться к шагу 2.

Оценка сложности алгоритма примерно равна $O((n-1) \cdot n^n \cdot m)$, так как трудно оценить из-за наличия бесконеного цикла в реализации данного алгоритма.

3.4 Коды программ, реализующей рассмотренные алгоритмы

```
import numpy as np
import math
```

```python
from itertools import product


def print_set(s, n):
    print('{', end=' ')
    k = 1
    for el in s:
        if k == n:
            print(el, '}')
        else:
            print(str(el) + ',', end=' ')
        k += 1


def create_subsemigroup():
    print('Enter set values:')
    s = input()
    set_list = [i for i in s.split(' ')]
    print('Enter Cayley table values:')
    c_tbl = []
    for i in range(len(set_list)):
      c_tbl.append([j for j in input().split()])
    print('Enter subset values:')
    s = input()
    subset_list = [i for i in s.split(' ')]
    new_subset = subset_list.copy()
    while True:
        tmp_set = []
        for el1 in set_list:
            for el2 in new_subset:
                tmp_set.append(c_tbl[set_list.index(el2)][set_list.index(el1)])
        subsemigroup = set(new_subset).union(set(tmp_set))
        if (subsemigroup == set(new_subset)):
            break
        else:
            new_subset = list(subsemigroup)

    subsemigroup = list(subsemigroup)
    subsemigroup.sort()
    print('Your subsemigroup:', end='')
    print_set(subsemigroup, len(subsemigroup))
```

```python
    choose_mode()


def create_bin_rel_semigroup():
    print('Enter elements of binary relation:')
    s = input()
    br_list = [i for i in s.split(' ')]
    n = len(br_list)
    print('Enter matrices dimension')
    d = int(input())
    matrices_dict = {}
    for i in range(n):
        print(f'Enter matrix values for binary relation \"{br_list[i]}\":')
        matrix = [list(map(int, input().split())) for i in range(d)]
        matrix = np.array(matrix).reshape(d, d)
        matrices_dict[br_list[i]] = matrix

    correlations = {}
    while True:
        new_matrices_dict = {}
        for key1, val1 in matrices_dict.items():
            for key2, val2 in matrices_dict.items():
                new_matrices_dict[key1 + key2] = val1 * val2
        check_set = set(br_list.copy())
        for key1, val1 in new_matrices_dict.items():
            fl = True
            for key2, val2 in matrices_dict.items():
                if (np.array_equal(val1, val2)) and key1 != key2:
                    correlations[key1] = key2
                    fl = False
                    break
            if fl:
                matrices_dict[key1] = val1
                br_list.append(key1)
        if check_set == set(br_list):
            break

    print("Your semigroup: ")
    print_set(br_list, len(br_list))

    print("Your semigroup (matrices): ")
```

```
    for key, value in matrices_dict.items():
        print(key, ":\n", value)

    print("Your correlations: ")
    for key, val in correlations.items():
        print(key + '->' + val)
    choose_mode()

def create_semigroup_via_set_simply():
    print('Enter elements of set:')
    s = input()
    set_list = [i for i in s.split(' ')]
    print('Number of elements in presentation:')
    k = int(input())
    presentation = {}
    for i in range(k):
        print(f'Enter element №{i + 1}')
        key = input()
        print(f'Enter equivalent of element №{i + 1}')
        val = input()
        presentation[key] = val
    semigroup = set_list.copy()
    while True:
        new_elements = []
        for el1 in semigroup:
            for el2 in semigroup:
                new_word = el1 + el2
                while True:
                    tmp = str(new_word)
                    for key, val in presentation.items():
                        if key in new_word:
                            new_word = new_word.replace(key, val)
                    if tmp == new_word:
                        break
                new_elements.append(new_word)
        check_semgr = set(semigroup.copy())
        for el in new_elements:
            if el not in semigroup:
                semigroup.append(el)
        if check_semgr == set(semigroup):
            break
```

```python
    print("Your semigroup:")
    print(semigroup)
    choose_mode()

# Главное меню
def choose_mode():
    print('Choose mode:')
    print('Press 1 to create subsemigroup')
    print('Press 2 to create binary relation semigroup')
    print('Press 3 to create semigroup via set')
    print('Press 4 to exit')
    bl = input()
    if bl == '1':
        create_subsemigroup()
    elif bl == '2':
        create_bin_rel_semigroup()
    elif bl == '3':
        create_semigroup_via_set_simply()
    elif bl == '4':
        return
    else:
        print('Incorrect output')
        return choose_mode()


choose_mode()
```

3.5 Результаты тестирования программ

На рисунке 1 показана работа алгоритма построения подполугруппы.

```
Choose mode:
Press 1 to create subsemigroup
Press 2 to create binary relation semigroup
Press 3 to create semigroup via set
Press 4 to exit
1
Enter set values:
a b c d
Enter Cayley table values:
a b a b
a b a b
a b c d
a b c d
Enter subset values:
a b
Your subsemigroup:{ a, b }
Choose mode:
Press 1 to create subsemigroup
Press 2 to create binary relation semigroup
Press 3 to create semigroup via set
Press 4 to exit
1
Enter set values:
a b c d
Enter Cayley table values:
a b a b
a b a b
a b c d
a b c d
Enter subset values:
a b c
Your subsemigroup:{ a, b, c, d }
Choose mode:
Press 1 to create subsemigroup
Press 2 to create binary relation semigroup
Press 3 to create semigroup via set
Press 4 to exit
4
```

Рисунок 1 – Тест алгоритма построения подполугруппы

На рисунках 2-4 изоражен результат работы алгоритма построения полугруппы с помощью булевых матриц.

```
Press 4 to exit
2
Enter elements of binary relation:
a b c
Enter matrices dimension
3
Enter matrix values for binary relation "a":
1 0 1
0 0 1
1 1 1
Enter matrix values for binary relation "b":
1 1 1
0 1 1
1 0 0
Enter matrix values for binary relation "c":
0 0 1
1 1 0
0 1 1
```

Рисунок 2 – Тест алгоритма построения полугруппы с помощью булевых матриц

```
Your semigroup:
{ a, b, c, ab, ac, bc, abc }
Your semigroup (matrices):
a :
 [[1 0 1]
 [0 0 1]
 [1 1 1]]
b :
 [[1 1 1]
 [0 1 1]
 [1 0 0]]
c :
 [[0 0 1]
 [1 1 0]
 [0 1 1]]
ab :
 [[1 0 1]
 [0 0 1]
 [1 0 0]]
ac :
 [[0 0 1]
 [0 0 0]
 [0 1 1]]
bc :
 [[0 0 1]
 [0 1 0]
 [0 0 0]]
abc :
 [[0 0 1]
 [0 0 0]
 [0 0 0]]
```

Рисунок 3 – Вывод полугруппы, выраженной матрицами бинарных отношений

```
Your correlations:
aa->a
ba->ab
bb->b
ca->ac
cb->bc
cc->c
aab->ab
aac->ac
bab->ab
bac->abc
bbc->bc
cab->abc
cac->ac
cbc->bc
aba->ab
abb->ab
abab->ab
abac->abc
abbc->abc
aca->ac
acb->abc
acc->ac
acab->abc
acac->ac
acbc->abc
bca->abc
bcb->bc
bcc->bc
```

Рисунок 4 – Вывод соотношений полугруппы (часть 1)

```
bcc->bc
bcab->abc
bcac->abc
bcbc->bc
aabc->abc
babc->abc
cabc->abc
ababc->abc
acabc->abc
bcabc->abc
abca->abc
abcb->abc
abcc->abc
abcab->abc
abcac->abc
abcbc->abc
abcabc->abc
Choose mode:
Press 1 to create subsemigroup
Press 2 to create binary relation semigroup
Press 3 to create semigroup via set
Press 4 to exit
```

Рисунок 5 – Вывод соотношений полугруппы (часть 2)

На рисунке 6 изоражен результат работы алгоритма построения полугруппы с помощью множества преобразований.

```
Choose mode:
Press 1 to create subsemigroup
Press 2 to create binary relation semigroup
Press 3 to create semigroup via set
Press 4 to exit
3
Enter elements of set:
x y
Number of elements in presentation:
3
Enter element №1
xy
Enter equivalent of element №1
yx
Enter element №2
xx
Enter equivalent of element №2
y
Enter element №3
yyy
Enter equivalent of element №3
x
Your semigroup:
['x', 'y', 'yx', 'yy', 'yyx']
Choose mode:
Press 1 to create subsemigroup
Press 2 to create binary relation semigroup
Press 3 to create semigroup via set
Press 4 to exit
4
```

Рисунок 6 – Тест алгоритма построения полугруппы с помощью множества преобразований

3.6 Решение задач

Задание 1. Найдите полугруппу S = $\langle f, g\rangle$ преобразований множества $X = 1, 2, 3$, порожденную следующими преобразованиями f, g в симметрической полугруппе $T(X)$ преобразований множества X:

$$f = \begin{pmatrix} 1 & 2 & 3 \\ 1 & 1 & 1 \end{pmatrix}, g = \begin{pmatrix} 1 & 2 & 3 \\ 2 & 3 & 2 \end{pmatrix}.$$

Известно, что множество преобразований f, g порождает полугруппу S = $\langle f, g\rangle$ преоб- разований множества X, которая состоит из элементов $f, g, f^2, fg, gf, g^2, \ldots$ и является подполугруппой конечной полугруппы $T(X)$.

$$f^2 = \begin{pmatrix}1&2&3\\1&1&1\end{pmatrix}\cdot\begin{pmatrix}1&2&3\\1&1&1\end{pmatrix} = \begin{array}{cccc} & 1 & 2 & 3\\ \text{f} & \downarrow & \downarrow & \downarrow \\ & 1&1&1\\ \text{f} & \downarrow&\downarrow&\downarrow\\ &1&1&1\end{array} = \begin{pmatrix}1&2&3\\1&1&1\end{pmatrix}$$

$$fg = \begin{pmatrix}1&2&3\\1&1&1\end{pmatrix}\cdot\begin{pmatrix}1&2&3\\2&3&2\end{pmatrix} = \begin{array}{cccc} & 1 & 2 & 3\\ \text{f} & \downarrow & \downarrow & \downarrow \\ & 1&1&1\\ \text{g} & \downarrow&\downarrow&\downarrow\\ &2&2&2\end{array} = \begin{pmatrix}1&2&3\\2&2&2\end{pmatrix}$$

$$gf = \begin{pmatrix}1&2&3\\2&3&2\end{pmatrix}\cdot\begin{pmatrix}1&2&3\\1&1&1\end{pmatrix} = \begin{array}{cccc} & 1 & 2 & 3\\ \text{g} & \downarrow & \downarrow & \downarrow \\ & 2&3&2\\ \text{f} & \downarrow&\downarrow&\downarrow\\ &1&1&1\end{array} = \begin{pmatrix}1&2&3\\1&1&1\end{pmatrix}$$

$$g^2 = \begin{pmatrix}1&2&3\\2&3&2\end{pmatrix}\cdot\begin{pmatrix}1&2&3\\2&3&2\end{pmatrix} = \begin{array}{cccc} & 1 & 2 & 3\\ \text{g} & \downarrow & \downarrow & \downarrow \\ & 2&3&2\\ \text{g} & \downarrow&\downarrow&\downarrow\\ &3&2&3\end{array} = \begin{pmatrix}1&2&3\\3&2&3\end{pmatrix}$$

Таким образом получаем полугруппу: $S = \langle f, g, fg, g^2, \dots \rangle$. Стоит отметить, что $gf \notin S$, так как $gf = f$.

Задание 2.

Найдите индекс и период следующих элементов a полугруппы преобразований множества $X = 1, 2, 3, 4, 5$

$$a = \begin{pmatrix}1&2&3&4&5\\2&4&1&2&1\end{pmatrix}$$

Посчитаем:

$$aa = \begin{array}{cccccc} & 1 & 2 & 3 & 4 & 5\\ \text{a} & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow \\ & 2&4&1&2&1\\ \text{a} & \downarrow&\downarrow&\downarrow&\downarrow&\downarrow\\ &4&2&2&4&2\end{array} = \begin{pmatrix}1&2&3&4&5\\4&2&2&4&2\end{pmatrix}$$

$$aaa = \begin{array}{cccccc} & 1 & 2 & 3 & 4 & 5 \\ \text{a} & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow \\ & 2 & 4 & 1 & 2 & 1 \\ \text{a} & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow \\ & 4 & 2 & 2 & 4 & 2 \\ \text{a} & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow \\ & 2 & 4 & 4 & 2 & 4 \end{array} = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 \\ 2 & 4 & 4 & 2 & 4 \end{pmatrix}$$

$$aaaa = \begin{array}{cccccc} & 1 & 2 & 3 & 4 & 5 \\ \text{a} & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow \\ & 2 & 4 & 1 & 2 & 1 \\ \text{a} & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow \\ & 4 & 2 & 2 & 4 & 2 \\ \text{a} & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow \\ & 2 & 4 & 4 & 2 & 4 \\ \text{a} & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow \\ & 4 & 2 & 2 & 4 & 2 \end{array} = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 \\ 4 & 2 & 2 & 4 & 2 \end{pmatrix}$$

Видно, что $aaaa \to aa$. Т.е. на 4 преобразовании наблюдается цикличность, тогда, если считать элементы полугруппы $\langle a, aa, aaa, aaaa, ...\rangle$, начиная с единицы, то каждый $2k$-й элемент будет иметь преобразование $\begin{pmatrix} 1 & 2 & 3 & 4 & 5 \\ 4 & 2 & 2 & 4 & 2 \end{pmatrix}$, а каждый $(2k+1)$-й элемент равен – $\begin{pmatrix} 1 & 2 & 3 & 4 & 5 \\ 2 & 4 & 4 & 2 & 4 \end{pmatrix}$, где $k \in \mathbb{N}$. Получается, что период будет равен 2.

Задание 3.

Найдите полугруппу S по следующему ее копредставлению:

$$S = \langle x, y : xy = yx, x^2 = y, y^3 = x\rangle$$

Выделим полную систему представителей классов конгруэнции ϵ, которая определяется соотношениями данного копредставления. Для этого последовательно рассмотрим слова фиксированной длины и выделим те, которые не будут эквивалентны между собой относительно конгруэнции ϵ.

Рассмотрим слова длины 1: x, y — эти слова не эквивалентны между собой относительно конгруэнции ϵ.

Рассмотрим слова длины 2, которые получаются из слов длины 1 путем последовательного умножения их справа на буквы x и y: $x^2 = y, xy, yx = xy, y^2$ – из этих слов только слова xy, y^2 не эквивалентны относительно конгруэнции ϵ другим ранее выделенным словам.

Теперь рассмотрим слова длины 3, которые получаются из выделенных слов длины 2 путем последовательного умножения их справа на буквы x и y: $xyx = y^2$, xy^2, $y^2x = x^2y$, $y^3 = x$ — из этих слов только слово xy^2 не эквивалентно относительно конгруэнции ε другим ранее выделенным словам.

Наконец рассмотрим слова длины 4, которые получаются из выделенного слова длины 3 путем последовательного умножения его справа на буквы x и y: $xy^2x = x^2y^2 = y^3 = x, xy^3 = x^2 = y$ - все эти слова эквивалентны относительно конгруэнции ε ранее выделенным словам.

Значит, $S = \{x, y, xy, y^2, xy^2\}$ — полная система представителей классов конгруэнции ε. Операция умножения $\cdot$ таких слов определяется с точностью до конгруэнции ε по следующей таблице Кэли:

$\cdot$	x	y	xy	y^2	xy^2
x	y	xy	y^2	xy^2	x
y	xy	y^2	xy^2	x	y
xy	y^2	xy^2	x	y	xy
y^2	xy^2	x	y	xy	y^2
xy^2	x	y	xy	y^2	xy^2

ЗАКЛЮЧЕНИЕ

В результате лабораторной работы были рассмотрены теоретические сведения о полугруппах, подполугруппах и порождающих множествах. Опираясь на изложенную выше теорию, были разработаны алгоритмы проверки свойств операций: ассоциативность, коммутативность, идемпотентность, обратимость, дистрибутивность, алгоритмы построения подполугруппы по таблице Кэли, построения полугруппы бинарных отношений по заданному порождающему множеству, построения полугруппы по порождающему множеству и определяющим соотношениям. Была произведена оценка сложности каждого из построенных алгоритмов. Была реализована программа, написанная на языке Python с использованием библиотеки Numpy, Math, Itertools для работы с большими массивами данных.