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

# КЛАССИФИКАЦИЯ БИНАРНЫХ ОТНОШЕНИЙ И СИСТЕМЫ ЗАМЫКАНИЙ

ЛАБОРАТОРНАЯ РАБОТА

студента 3 курса 331 группы
специальности 100501 — Компьютерная безопасность
факультета КНиИТ
Окунькова Сергея Викторовича

Проверил
аспирант ______________ В. Н. Кутин

Саратов 2022

**СОДЕРЖАНИЕ**

## 1 Постановка задачи

Цель работы:

Изучение основных свойств бинарных отношений и операций замыкания бинарных отношений.

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

**Полугруппа** – это алгебра $S = (S, \cdot)$ с однойассоциативной бинарной операцией $\cdot$, т.е. выполняется $(x \cdot y) \cdot z = x \cdot (y \cdot z)$ для любых $x, y, z \in S$.

Если полугрупповая операция называется умножением (соответственно, сложением), то полугруппу называют мультипликативной (соответственно, аддитивной).

Подмножество X полугруппы S называется **подполугруппой**, если X устойчиво относительно операции умножения, т.е. для любых $x, y \in X$ 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 $S$.

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

Для любой конечной полугруппы S найдется такой конечный алфавит A, что для некоторого отображения $\phi : A \to S$ выполняется равенство $< \phi(A) >= S$ 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 $\rho$, 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 $u, v \in A^+$ :

(а) 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 $x, y \in A^*$;

(б) слово $v$ выводится из слова $u$, если $v$ получается из $u$ с помощью конечного числа применения процедуры (а).

Если все выполняющиеся на $S$ соотношения выводятся из определяющих соотношений совокупности $\rho$, то конгруэнция $ker\phi$ полностью определяется отношением $\rho$ 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 $X_{i+1} = X_i \cup \overline{X}_l$ (выражение $x \cdot y$ означает $a_{xy}$ в таблице Кэли $A$).

Шаг 3. Если $X_{i+1} = X_i$ вернем $X_i$, которое будет являться подполугруппой $\langle X \rangle \subset S$, иначе положим $i = i + 1$ и вернемся ко 2-му шагу.

Трудоемкость алгоритма $O(n^3)$.

### 3.2 Алгоритм 2 - Построение полугруппы бинарных отношений по заданному порождающему множеству

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

*Выход*: Полугруппа $\langle X \rangle$.

Шаг 1. Необходимо инициализировать список $cur = X$, в нем будут хранится матрицы заданных бинарных отношений.

Шаг 2. Необходимо найти поэлементное произведение каждой матрицы из списка $cur$ на каждую матрицу из списка $X$ $(\forall a \in cur, \forall b \in X a \circ b)$ и проверить есть ли полученные матрицы в $X$.

Шаг 3. Если все полученные матрицы уже лежат в $X$, вернуть $X$, иначе добавить матрицы бинарных отношений, который не принадлежат $X$ в $X$ и вернуться к шагу 2.

Трудоемкость алгоритма $O(m^n)$.

### 3.3 Алгоритм 3 - Построение полугруппы по порождающему множеству и определяющим соотношениям

*Вход*: Конечное множество символов $A$ размерности $n$ и конечное множество $R$ определяющих соотношений размерности $m \times n$.

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

Шаг 1. Необходимо инициализировать пустой список $gl$ и добавить в него все элементы генератора $R$. Инициализировать массив $alph = gl$

<u>Шаг 2.</u> К каждому элемента $gl$ добавить каждый элемент $alph$ и посчитать новое определяющее соотношение. Оно будет рассчитываться как массив, размерности $n$, где k элемент $(0 \leq k < n)$ будет равен $R[i][j]$, где i - это добавленный элемент из $alph$, а j - позиция k-го элемента в $A$.

<u>Шаг 3.</u> Если все полученные значения соотношений уже есть в $R$, вернуть $R$, иначе $gl =$ все уникальные соотношения и вернуться к шагу 2.

Трудоемкость алгоритма $O(m^n)$.

### 3.4 Коды программ, реализующей рассмотренные алгоритмы

```
import numpy as np

def add_correlation(cur_word, alph, dct, semigroup_elems):
    new_words = []
    for letter in alph:
        new_word = cur_word + letter
        m = []
        for i in dct[cur_word]:
            if i != '*':
                m.append(dct[letter][semigroup_elems.index(i)])
            else:
                m.append('*')
        flag = True
        for key in dct:
            if m == dct[key]:
                print(new_word, '->', key)
                flag = False
        if flag:
            dct[new_word] = m
            new_words.append(new_word)
    return new_words


def add_coopresentation(cur_bin_relations, bin_relations, k):
    dct = bin_relations.copy()
    flag = False
    for key1 in cur_bin_relations:
        for key in bin_relations:
            if k == len(key):
```

```
                bin_relation = cur_bin_relations[key1] * bin_relations[key]
                new_key = key + key1
                f = True
                for key2 in dct:
                    if np.array_equal(bin_relation, dct[key2]):
                        print(new_key, "->", key2)
                        f = False
                        break
                if f:
                    dct[new_key] = bin_relation
                    flag = True
    return flag, dct


def task1():
    print("Enter your set")
    st = list(input().split())
    print('Enter Cayley table')
    print("   ", *st)
    matrix = []
    for i in range(len(st)):
        print(st[i], end="   ")
        s = list(input().split())
        matrix.append(s)
    print("Enter your subset")
    subst = list(input().split())
    x_i = subst.copy()
    while True:
        x_l = []
        for x in x_i:
            for y in subst:
                x_l.append(matrix[st.index(x)][st.index(y)])
        x_0 = x_i.copy()
        x_0.sort()
        x_i = list(set(x_i).union(set(x_l)))
        x_i.sort()
        if x_0 == x_i:
            print('Subsemigroup is', *x_i)
            break
```

```
def task2():
    print("Enter the elements of the set: ")
    input_list = input().replace(",", "").split()
    n = len(input_list)
    print("Enter the number of binary relations")
    bin_relation_amount = int(input())
    bin_relation_matrices = {}
    for i in range(1, bin_relation_amount + 1):
        print(f"Enter boolean matrix Values {i} binary relation: ")
        print(" ", *input_list)
        matrix = [list(map(int, input(f"{input_list[i]} ").split())) for i in range(n)
        matrix = np.array(matrix).reshape(n, n)
        bin_relation_matrices[str(i)] = matrix

    flag = True
    cur = bin_relation_matrices.copy()
    k = 1
    print("The resulting ratios: ")
    while flag:
        flag, bin_relation_matrices = add_coopresentation(cur, bin_relation_matrices,
        k += 1
    for key in bin_relation_matrices:
        print(key, ':')
        print(bin_relation_matrices[key])


def task3():
    print("Enter semigroup elements: ")
    semigroup_elems = [elem for elem in input().replace(",", "").split()]
    print("Enter the elements of the transformation set: ")
    generators_list = input().replace(",", "").split()
    translation_dict = {}
    for i in range(len(generators_list)):
        print(f"Enter transformation values '{generators_list[i]}' elements of the se
        print((str(semigroup_elems)[1:-1]).replace(", ", " ").replace("'", ""))
        translation = input().split()
        translation_dict[generators_list[i]] = translation
    print("Coopresentation: ")
    gl = generators_list.copy()
    while gl:
        gl1 = []
```

```python
        for s in gl:
            gl1.extend(add_correlation(s, generators_list, translation_dict, semigrou
        gl = gl1
    print("The resulting ratios: ")
    print(translation_dict)

if __name__ == "__main__":
    print("What are you want? (1 - Build subsemigroup by Cayley table, 2 - Build semi
    f = int(input())
    if f == 1:
        task1()
    elif f == 2:
        task2()
    elif f == 3:
        task3()
    else:
        print("Something going wrong! Enter a number from 1 to 3")
```

### 3.5 Результаты тестирования программ

```
What are you want? (1 - Build subsemigroup by Cayley table, 2 - Build semigroup binary relations, 3 - Build semigroup by generating set and transformation set)
1
Enter your set
1 2 3
Enter Cayley table
   1 2 3
1  2 3 1
2  3 1 2
3  1 2 3
Enter your subset
2 3
Subsemigroup is 1 2 3
```

Рисунок 1 – Тест алгоритма построения подполугрупп по по таблице Кэли

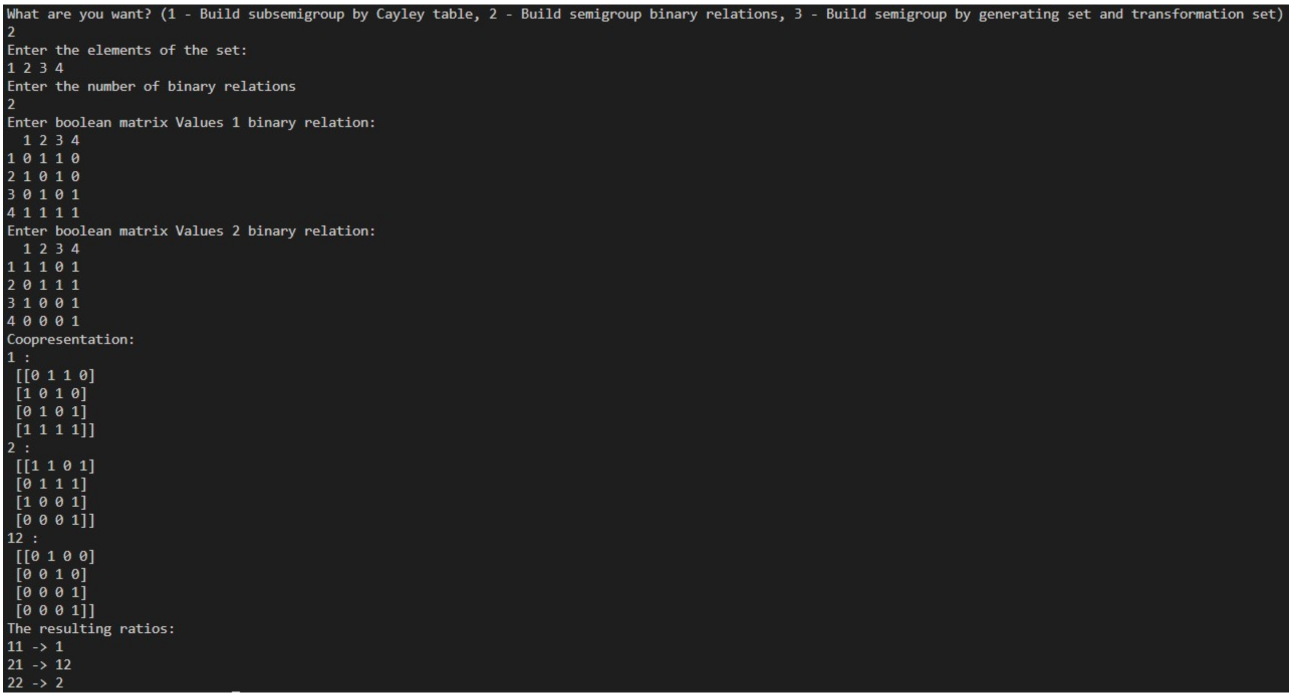


Рисунок 2 – Тест алгоритма построения полугруппы бинарных отношений по заданному порождающему множеству

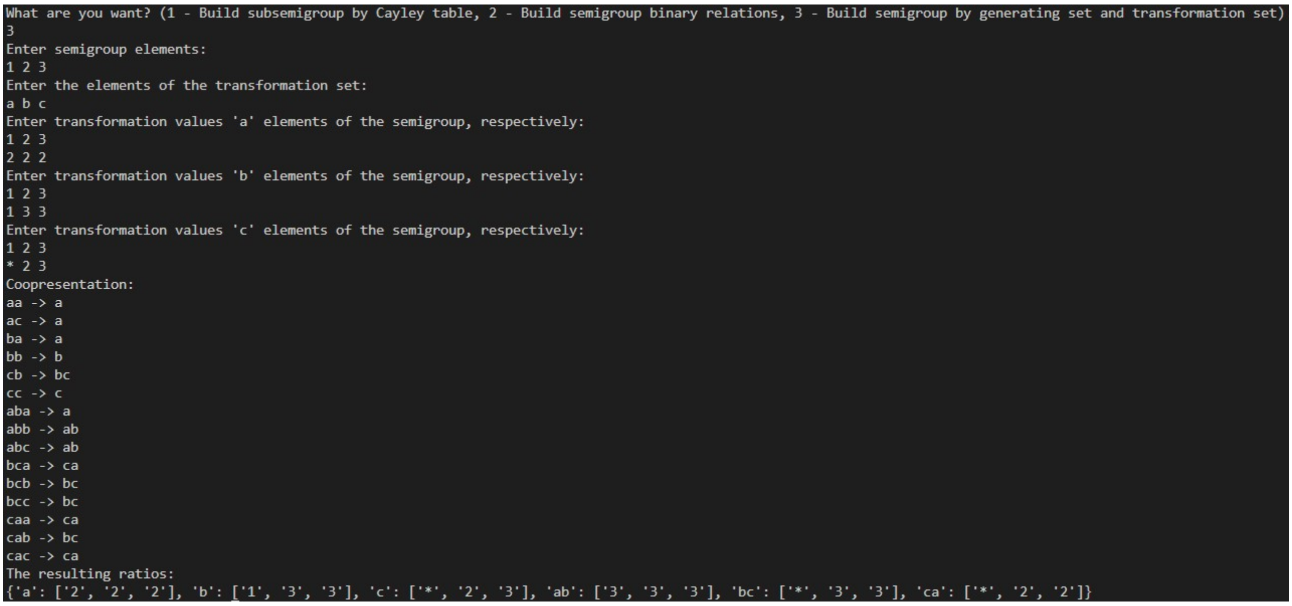


Рисунок 3 – Тест алгоритма построения полугруппы по порождающему множеству и определяющим соотношениям

### 3.6 Ответы на задачи

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

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

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

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

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

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

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

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

**Задание 2.** Найдите индекс и период следующих элементов a полугруппы преобразований множества X=1,2,3,4,5:

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

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

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

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

Видно, что $aaaa \to aa$. Т. е. период будет равен 2. **Задание 3.** Найдите полугруппу S по ее копредставлению $\langle x, y : xy = yx, x^3 = x^2, y^2 = y\rangle$. Выделим полную систему представителей классов конгруэнции $\varepsilon$, которая определяется соотношениями данного копредставления. Для этого последовательно рассмотрим слова фиксированной длины и выделим те, которые не будут эквивалентны между собой относительно конгруэнции $\varepsilon$.

Сначала рассматриваем слова длины 1: $x, y$ - эти слова не эквивалентны между собой относительно конгруэнции $\varepsilon$.

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

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

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

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

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

# ЗАКЛЮЧЕНИЕ

В рамках данной лабораторной работы были рассмотренны теоритические основы свойств бинарных подгруп и полугрупп, а также способы их построения. На основе этой теоретической части была смоделирована программа на языке Python с использованием средств библиотеки Numpy, которая способна построить подполугрупп по по таблице Кэли, построить полугруппу бинарных отношений по заданному порождающему множеству и построить полугруппу по порождающему множеству и определяющим соотношениям.