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.

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 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 $u, v \in A^+$:

(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 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 $X_{i+1} = X_i \cup \overline{X}_l$ (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, заданное булевыми матрицами размерности $n \times n$.

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 $\langle X \rangle$.

Шаг 2. Необходимо создать список $combinations$, элементы которого будут $c_k \in combinations$, где $0 \leq k < (n^1 + n^2 + ... + n^n)$. Т.е. этот список является суммой размещений с повторениями.

Шаг 3. Далее возьмем матрицу A_i $(0 \leq i < n)$ и умножим ее на матрицы $B_0, ..., B_l$ согласно текущей комбинации c_k $(0 \leq k < (n^1 + n^2 + ... + n^n))$, где матрицы $B_1, ..., B_l \in M$ составляют текущую комбинацию c_k (l – количество элементов в c_k). Таким образом получаем матрицу $C = A_i \odot B_1 \odot \cdots \odot B_l$, где $\odot$ – операция поэлементного умножения. Добавляем C в список $matrices$ в качестве нового элемента полугруппы $\langle X \rangle$.

Шаг 4. Повторять шаг 3 k раз $(0 \leq k < (n^1 + n^2 + ... + n^n))$.

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 $alph = gl$

Шаг 2. 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 $alph$, a j - позиция k-го элемента в A.

Шаг 3. 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
from itertools import product

def get_set(a):
    s = []
    for i in range(len(a)):
        for j in range(len(a[i])):
            if a[i][j] == 1:
                s.append((i + 1, j + 1))
    return s


def subsets(n):
    nums = [i for i in range(n)]
    from functools import reduce
    return reduce(lambda res, x: res + [subset + [x] for subset in res],
                  nums, [[]])[1:]
```

```
def get_res(matrixes):
    res = []
    for subset in subsets(len(matrixes)):
        podres = matrixes[subset[0]]
        for index in subset[1:]:
            podres *= matrixes[index]
        for relation in get_set(podres):
            res.append(relation)
    return set(res)


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 find_correlation(ans):
    result = {}
    correlations = {}
    for key, value in ans.items():
        if not any(np.array_equal(value, i) for i in result.values()):
            result[key] = value
        else:
            for k, v in result.items():
```

```
                if np.array_equal(v, value):
                    correlations[key] = k
    print("Coopresentation: ")
    for key, value in result.items():
        print(key, ":\n", value)

    print("The resulting ratios: ")
    for key, value in correlations.items():
        print(key, "->", value)


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[x_i.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

    combinations_list = []
    for i in range(1, bin_relation_amount + 1):
        combinations = list(product(''.join([str(elem) for elem in range(1, bin_relat
        combinations_list += combinations

    for comb in combinations_list:
        cur_matrix = bin_relation_matrices[comb[0]].copy()
        word = comb[0]
        for comb_i in range(1, len(comb)):
            cur_matrix *= bin_relation_matrices[comb[comb_i]]
            word += comb[comb_i]
        bin_relation_matrices[word] = cur_matrix

    find_correlation(bin_relation_matrices)


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 = []
        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

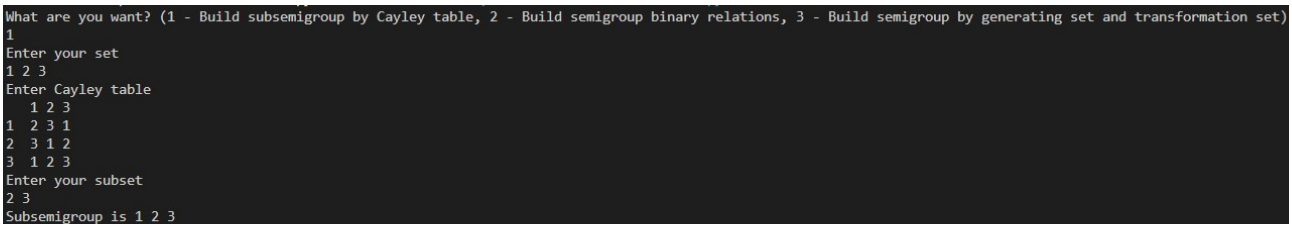

Рисунок 1 – Тест алгоритма построения подполугрупп по по таблице Кэли

```
What are you want? (1 - Build subsemigroup by Cayley table, 2 - Build semigroup binary relations, 3 - Build semigroup by generating set and transformation set)
2
Enter the elements of the set:
1 2 3 4
Enter the number of binary relations
2
Enter boolean matrix Values 1 binary relation:
  1 2 3 4
1 0 1 1 0
2 1 0 1 0
3 0 1 0 1
4 1 1 1 1
Enter boolean matrix Values 2 binary relation:
  1 2 3 4
1 1 1 0 1
2 0 1 1 1
3 1 0 0 1
4 0 0 0 1
Coopresentation:
1 :
 [[0 1 1 0]
 [1 0 1 0]
 [0 1 0 1]
 [1 1 1 1]]
2 :
 [[1 1 0 1]
 [0 1 1 1]
 [1 0 0 1]
 [0 0 0 1]]
12 :
 [[0 1 0 0]
 [0 0 1 0]
 [0 0 0 1]
 [0 0 0 1]]
The resulting ratios:
11 -> 1
21 -> 12
22 -> 2
```

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

```
What are you want? (1 - Build subsemigroup by Cayley table, 2 - Build semigroup binary relations, 3 - Build semigroup by generating set and transformation set)
3
Enter semigroup elements:
1 2 3
Enter the elements of the transformation set:
a b c
Enter transformation values 'a' elements of the semigroup, respectively:
1 2 3
2 2 2
Enter transformation values 'b' elements of the semigroup, respectively:
1 2 3
1 3 3
Enter transformation values 'c' elements of the semigroup, respectively:
1 2 3
* 2 3
Coopresentation:
aa -> a
ac -> a
ba -> a
bb -> b
cb -> bc
cc -> c
aba -> a
abb -> ab
abc -> ab
bca -> ca
bcb -> bc
bcc -> bc
caa -> ca
cab -> bc
cac -> ca
The resulting ratios:
{'a': ['2', '2', '2'], 'b': ['1', '3', '3'], 'c': ['*', '2', '3'], 'ab': ['3', '3', '3'], 'bc': ['*', '3', '3'], 'ca': ['*', '2', '2']}
```

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

3.6 Ответы на задачи

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:

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

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 а 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 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.

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

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 ε 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 ε 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 ε ранее выделенным словам.

Значит, $S = \{x, y, y^2, xy, xy^2\}$ — 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	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.