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

ИДЕАЛЫ 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 строения полугрупп с помощью отношений Грина.

Порядок выполнения работы:

1. Рассмотреть понятия идеалов полугруппы. Разработать алгоритмы построения идеалов полугруппы по таблице Кэли.
2. Рассмотреть понятия и свойства отношений Грина на полугруппах.
3. Разработать алгоритмы вычисления отношений Грина и построения «egg-box»-картины конечной полугруппы.

2 Теоретические сведения по рассмотренным темам с их обоснованием

Пусть S – произвольная полугруппа.

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

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

для любых $x, y, z \in S$.

Определение 2. Непустое подмножество $I \subset S$ называется правым (левым) идеалом полугруппы S, если для любых $x \in I, y \int S$ выполняется условие: $xy \int I$ ($yx \in I$), т.е. $I \cdot S \subset I$ ($S \cdot I \subset I$). Если I – одновременно левый и правый идеал полугруппы S, то I называется двусторонним идеалом (или просто идеалом) полугруппы S. Ясно, что в коммутативной полугруппе S все эти определения совпадают.

Лемма 1. Множество всех идеалов IdS (соответственно, левых идеалов $LIdS$ или правых идеалов $RIdS$) любой полугруппы S является системой замыкания. Пусть X – подмножество полугруппы S. Тогда наименьший правый идеал полугруппы S, содержащий подмножество X, равен $(X] = XS^1 = X \cup XS$, наименьший левый идеал полугруппы S, содержащий подмножество X, равен $[X) = S^1X = X \cup SX$ и наименьший идеал полугруппы S, содержащий подмножество X, равен $[X] = S^1XS^1 = X \cup XS \cup SX \cup SXS$.

В частности, любой элемент $a \in S$ определяет наименьшие правый, левый и двусторонний идеалы: $(a] = aS^1$, $[a) = S^1a$ и $[a] = S^1aS^1$, которые называются главными (соответственно, правыми, левыми и двусторонними) идеалами. Минимальные относительно теоретико-множественного включения идеалы (левые или правые идеалы) называются минимальными идеалами (минимальными левыми или правыми идеалами).

Лемма 2. Если полугруппа имеет минимальный идеал, то он является ее наименьшим идеалом и называется ядром полугруппы.

Пример: В полугруппе натуральных чисел с операцией сложения $\mathbf{N} = (\mathbf{N}, +)$ главные идеалы $(n] = n, n+1, n+2, \ldots$ образуют бесконечную последовательность с пустым пересечением.

Отображения $f : a \mapsto [a]$, $f_r : a \mapsto (a]$, $f_l : a \mapsto [a)$, $a \in S$ определяют ядра $\mathfrak{J} = kerf$, $\mathfrak{R} = kerf_r$, $\mathfrak{L} = kerf_l$ по формулам:

$$(a,b) \in \mathfrak{J} \Longleftrightarrow [a] = [b],$$
$$(a,b) \in \mathfrak{R} \Longleftrightarrow (a] = (b],$$
$$(a,b) \in \mathfrak{L} \Longleftrightarrow [a) = [b).$$

Все эти отношения, а также отношения $\mathfrak{D} = \mathfrak{R} \vee \mathfrak{L}$, $\mathfrak{H} = \mathfrak{R} \cap \mathfrak{L}$ являются эквивалентностями на множестве S, которые называются **отношениями Грина** полугруппы S. Классы этих эквивалентностей, порожденные элементом $a \in S$, обозначаются J_a, R_a, L_a, D_a и H_a, соответственно.

Лемма 3. Отношения Грина полугруппы S удовлетворяют следующим свойствам:

1. эквивалентность $\mathfrak{R}$ регулярна слева и эквивалентность $\mathfrak{L}$ регулярна справа, т.е. $(a,b) \in \mathfrak{R} \Rightarrow (xa, xb) \in \mathfrak{R}$ и $(a,b) \in \mathfrak{L} \Rightarrow (ax, bx) \in \mathfrak{L}$ для любых $x \in S$;
2. эквивалентности $\mathfrak{R}$, $\mathfrak{L}$ коммутируют;
3. $\mathfrak{D} = \mathfrak{R} \cdot \mathfrak{L} = \mathfrak{L} \cdot \mathfrak{R}$;
4. если полугруппа S конечна, то $\mathfrak{D} = \mathfrak{J}$;
5. любой класс $\mathfrak{D}$ эквивалентности $\mathfrak{D}$ можно изобразить с помощью следующей следующей «egg-box»-диаграммы, клетки которой являются классами эквивалентности $\mathfrak{H}$, лежащими в $\mathfrak{D}$.

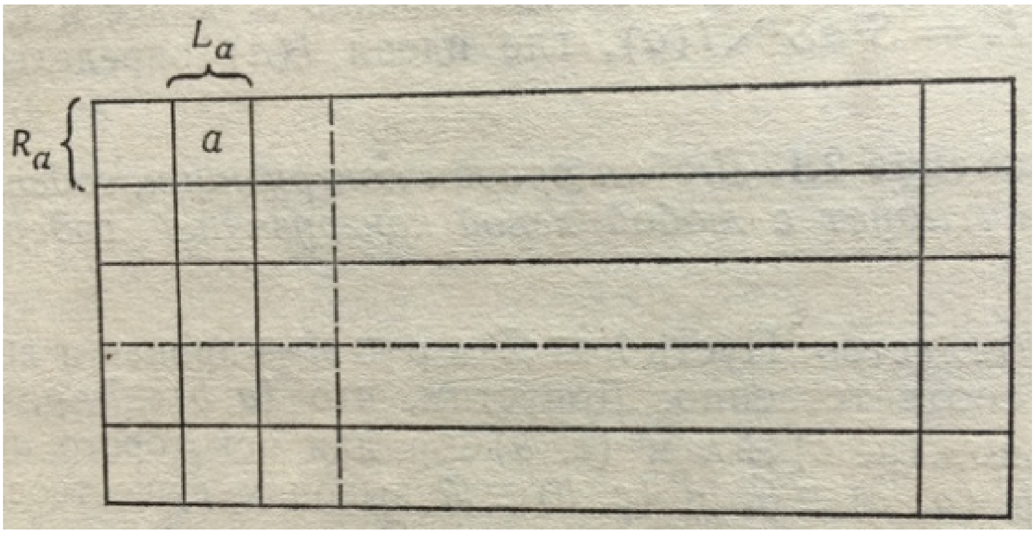

Рисунок 1 – «egg-box»-диаграмма

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. .

Шаг 2. .

Шаг 3. .

3.2 Коды программ, реализующей рассмотренные алгоритмы

```
import numpy as np
import math
from itertools import product


order = []
component = []

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


# Проверка операции на ассоциативность
def check_associative(set_list, a):
  n = len(set_list)
  for i in range(n):
    for j in range(n):
      for k in range(n):
        if a[i][set_list.index(str(a[j][k]))] != \
          a[set_list.index(str(a[i][j]))][k]:
```

```
        return False
  return True


def get_right_ideal(x, set_list, c_tbl):
    right_ideal = set()
    indx = set_list.index(x)
    for el in c_tbl[indx]:
        right_ideal.add(el)
    return right_ideal


def get_left_ideal(x, set_list, c_tbl):
  left_ideal = set()
  indx = set_list.index(x)
  for i in range(len(c_tbl)):
    left_ideal.add(c_tbl[i][indx])
  return left_ideal


def dfs(gr, visited, v):
    visited[v] = True
    for i in range(len(gr)):
        u = i
        if (not(visited[u]) and gr[v][u]):
            dfs(gr, visited, u)
    order.append(v)


def dfs1(t_gr, visited, v):
    visited[v] = True
    component.append(v)
    for i in range(len(t_gr)):
        u = i
        if (not(visited[u]) and t_gr[v][u]):
            dfs1(t_gr, visited, u)


def print_egg_boxes(semigroup, egg_box):
    print('Your egg-box-diagram:')
    print('{* 1 }')
```

```python
    for box in egg_box:
        print('{*', end=' ')
        for i in range(len(box)):
            print(semigroup[box[i]], end=' ')
        print('}')


def get_egg_boxes(semigroup, d):
    n = len(d)
    visited = [False for _ in range(n)]
    for i in range(n):
        if (not(visited[i])):
            dfs(d, visited, i)
    visited = [False for _ in range(n)]
    egg_box = []
    for i in range(n):
        v = order[n - 1 - i]
        if (not(visited[v])):
            dfs1(d.T, visited, v)
            egg_box.append(component.copy())
            component.clear()
    order.clear()
    return egg_box


def create_Grin_relation(semigroup, right_ideals_dict, left_ideals_dict):
    r = []
    l = []
    for el1 in semigroup:
        tmp_a = []
        tmp_b = []
        for el2 in semigroup:
            if right_ideals_dict[el1] == right_ideals_dict[el2]:
                tmp_a.append(1)
            else:
                tmp_a.append(0)
            if left_ideals_dict[el1] == left_ideals_dict[el2]:
                tmp_b.append(1)
            else:
                tmp_b.append(0)
        r.append(tmp_a)
```

```
            l.append(tmp_b)
    n = len(semigroup)
    # d = []
    # for i in range(n):
    #     tmp_a = []
    #     for j in range(n):
            # if r[i][j] == l[i][j] == 1:
            #     tmp_a.append(1)
            # else:
            #     tmp_a.append(r[i][j] + l[i][j])
    #     d.append(tmp_a)
    d = np.zeros((n, n))
    for i in range(n):
        for j in range(n):
            if r[i][j] == l[i][j] == 1:
                d[i][j] = 1
            else:
                d[i][j] = r[i][j] + l[i][j]

    print('Your Grin\'s relation:')
    for el in d:
        print(el)

    print('Would you like to get egg-box-diagram? (1 - "yes")')
    bl = input()
    if bl == '1':
        egg_box_list = get_egg_boxes(semigroup, d)

    print_egg_boxes(semigroup, egg_box_list)

def get_and_set_ideals(semigroup, c_tbl):
    right_ideals_dict = {}
    left_ideals_dict = {}
    for el in semigroup:
        print('-----------------------------------------')
        print(f'Right ideal ({el}]:', end=' ')
        right_ideals_dict[el] = get_right_ideal(el, semigroup, c_tbl)
        tmp_set = list(right_ideals_dict[el])
        tmp_set.sort()
        print_set(tmp_set)
        print(f'Left ideal [{el}):', end=' ')
```

```
        left_ideals_dict[el] = get_left_ideal(el, semigroup, c_tbl)
        tmp_set = list(left_ideals_dict[el])
        tmp_set.sort()
        print_set(tmp_set)
        print(f'Ideal [{el}]:', end=' ')
        tmp_set = list(left_ideals_dict[el].union(right_ideals_dict[el]))
        tmp_set.sort()
        print_set(tmp_set)
    print('-----------------------------------------')
    return right_ideals_dict, left_ideals_dict


def create_ideals():
    print('Enter set values:')
    s = input()
    semigroup = [i for i in s.split(' ')]
    n = len(semigroup)
    print('Enter Cayley table values:')
    c_tbl = []
    for i in range(n):
      c_tbl.append([j for j in input().split()])

    if check_associative(semigroup, c_tbl) == False:
        print('Cayley table isn\'t associative!')
        return choose_mode()

    right_ideals_dict, left_ideals_dict = get_and_set_ideals(semigroup, c_tbl)

    print('Would you to create Grin\'s relation? (1 - "yes")')
    bl = input()
    if bl == '1':
        create_Grin_relation(semigroup, right_ideals_dict, left_ideals_dict)
    return choose_mode()


def create_table(semigroup, n, presentation):
    a = []
    for i in range(n):
        tmp_a = []
        for j in range(n):
            new_word = semigroup[i] + semigroup[j]
```

```
            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
            tmp_a.append(new_word)
        a.append(tmp_a)
    return a


def create_semigroup_via_subset():
    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 = 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

    print("Your semigroup:")
    print(semigroup)
    tbl = create_table(semigroup, len(semigroup), presentation)
    print('Cayley table:')
    for line in tbl:
        print(line)

    right_ideals_dict, left_ideals_dict = get_and_set_ideals(semigroup, tbl)

    print('Would you to create Grin\'s relation? (1 - "yes")')
    bl = input()
    if bl == '1':
        create_Grin_relation(semigroup, right_ideals_dict, left_ideals_dict)

    choose_mode()


# Главное меню
def choose_mode():
    print('Choose mode:')
    print('Press 1 to create Grin\'s relations')
    print('Press 2 to create Grin\'s relations via subset')
    print('Press 3 to exit')
    bl = input()
    if bl == '1':
        create_ideals()
    elif bl == '2':
        create_semigroup_via_subset()
    elif bl == '3':
        return
    else:
        print('Incorrect output')
        return choose_mode()


choose_mode()
```

3.3 Результаты тестирования программ

На рисунке 1 показана работа алгоритма построения подполугруппы.

3.4 Решение задач

Задание 1. Найдите подполугруппу $\langle x \rangle$, правый $(x]$, левый $[x)$ и двусторонний $[x]$ идеалы полугруппы S, порожденные элементом x, и определите порядок элемента x для каждого элемента полугруппы, на которой бинарная операция задана следующей таблицей Кэли:

·	a	b	c	d
a	a	b	c	d
b	b	d	a	c
c	a	b	d	b
d	d	a	b	c

1. Нахождение подполугруппы:
   Подполугруппа строится по порождающему ее множеству. Допустим у нас есть подмножество $X \subset S$, где $X = \{a\}$. Тогда построим подполгруппу:
   Элемент a подмножества X определен в первой строчке таблицы Кэли. Поэтому мы должны пройти по элементам, находящихся в первой строке. Если еще такого элемента нет в подмножестве X, то он добавляется в данное подмножество. Т.е. пройдя по первой строке в данном случае получается подмножество $X = \{a, b, c, d\}$. Далее необходимо пройтись по строчкам, в котором определены новые элементы подмножества X (т.е. b, c, d). Рассмотрим элемент b и, пройдясь по второй строчке, видно, что новых элементов в подмножество X не добавилось. Значит, мы получили подполугруппу $\langle X \rangle = \{a, b, c, d\}$. Аналогично строится подполугруппа для b, c, d полугруппы S:
   Пусть $X \subset S$, где $X = \{b\}$. Тогда $\langle X \rangle = \{a, b, c, d\}$.
   Пусть $X \subset S$, где $X = \{c\}$. Тогда $\langle X \rangle = \{a, b, c, d\}$.
   Пусть $X \subset S$, где $X = \{d\}$. Тогда $\langle X \rangle = \{a, b, c, d\}$.
2. Нахождение идеалов:
   Используя таблицу Кэли построим правые идеалы:

$$(a] = \{a, b, c, d\}$$

$$(b] = \{a, b, c, d\}$$
$$(c] = \{a, b, d\}$$
$$(d] = \{a, b, c, d\}$$

Теперь построим левые идеалы:

$$[a) = \{a, b, d\}$$
$$[b) = \{a, b, d\}$$
$$[c) = \{a, b, c, d\}$$
$$[d) = \{b, c, d\}$$

Также построим двусторонние идеалы:

$$[a] = \{a, b, c, d\}$$
$$[b] = \{a, b, c, d\}$$
$$[c] = \{a, b, c, d\}$$
$$[d] = \{a, b, c, d\}$$

Задание 2.

Найдем отношения Грина для полугруппы $S = \{a, b, c, d\}$ из задания 1:

Заполним матрицу $\mathfrak{R}$, элементы которой будут определяться следующим образом: Возьмем правый идеал $(a]$ и рассмотрим относительно него остальные правые идеалы. Если, например, $(a] = (b]$, то на месте пересечения элементов a и b в матрице $\mathfrak{R}$ будет стоять 1, в противном случае будет стоять 0.

Тогда матрица будет выглядеть следующим образом:

$$\mathfrak{R} = \begin{pmatrix} 1 & 1 & 0 & 1 \\ 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 \end{pmatrix}$$

Аналогично построим матрицу $\mathfrak{L}$ по левым идеалам:

$$\mathfrak{L} = \begin{pmatrix} 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Тогда отношение Грина будет представлено матрицей $\mathfrak{D} = \mathfrak{R} \oplus \mathfrak{L}$:

$$\mathfrak{D} = \begin{pmatrix} 1 & 1 & 0 & 1 \\ 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 \end{pmatrix}$$

«Egg-box»-картина будет выглядеть следующим образом:

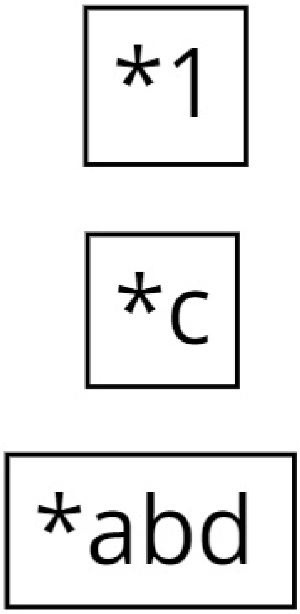

Рисунок 2 – «Egg-box»-картина

Задание 3.

Найдите полугруппу S по следующему ее копредставлению:

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

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: $x^2, xy, yx = xy, y^2 = x$

– из этих слов только слова x^2, xy, 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: $x^3 = x$, x^2y, xyx, $xy^2 = x^2$ — из этих слов только слово x^2y 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: $x^3y = xy$, $x^2y^2 = x^3 = x$ – все эти слова эквивалентны относительно конгруэнции ε ранее выделенным словам.

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

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

Соответственно правые идеалы будут иметь следующие значения:

$$(x] = \{x, x^2, xy, x^2y\}$$
$$(y] = \{x, x^2, xy, x^2y\}$$
$$(x^2] = \{x, x^2, xy, x^2y\}$$
$$(xy] = \{x, x^2, xy, x^2y\}$$
$$(x^2y] = \{x, x^2, xy, x^2y\}$$

Соответственно левые идеалы:

$$[x) = \{x, x^2, xy, x^2y\}$$
$$[y) = \{x, x^2, xy, x^2y\}$$
$$[x^2) = \{x, x^2, xy, x^2y\}$$
$$[xy) = \{x, x^2, xy, x^2y\}$$
$$[x^2y) = \{x, x^2, xy, x^2y\}$$

Тогда отношение Грина:

$$\mathfrak{D} = \mathfrak{R} \vee \mathfrak{L} = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{pmatrix} + \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{pmatrix} = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{pmatrix}$$

ЗАКЛЮЧЕНИЕ

А?