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.

Порождающее множество группы G (или множество образующих, или система образующих) — это подмножество S в G, такое, что каждый элемент G может быть записан как произведение конечного числа элементов S и их обратных.

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.

3 Результаты работы

3.1 Описание алгоритма классификации бинарных отношений

1. Алгоритм 1 - Проверка бинарного отношения на рефлексивность:
   Бинарное отношение называется рефлексивным тогда и только тогда, когда $\Delta_A \subset \rho$. Это означает, что бинарное отношение ρ рефлексивно, если $M(\rho) \geq E$, где E - единичная матрица. Если же матрица $M(\rho)$ несравнима с единичной матрицей, то бинарное отношение ρ не является рефлексивным;
   Вход: матрица бинарного отношения $A = (a_{ij})$, размерности $n \times n$
   Выход: "Множество рефлексивно"или "Множество не рефлексивно"
   Шаг 1. Суммирование элементов на главной диагонали ($sum = \sum\limits_{i=1}^{n} a[i][i]$).
   Шаг 2. Если $sum = n$, то отношение является рефлексивным, иначе не рефлексивным.
   Асимптотика $O(n)$.
2. Алгоритм 2 - Проверка бинарного отношения на симметричность:
   Бинарное отношение называется симметричным тогда и только тогда, когда $\rho^{-1} \subset \rho$. Это означает, что бинарное отношение ρ симметрично, если $M(\rho) \geq M(\rho)^T$, где $M(\rho)^T$ – транспонированная матрица бинарного отношения ρ. Если же матрица $M(\rho)$ несравнима с $M(\rho)^T$, то бинарное отношение ρ не является симметричным;
   Вход: матрица бинарного отношения $A = (a_{ij})$, размерности $n \times n$
   Выход: "Множество симметрично"или "Множество не симметрично"
   Шаг 1. Транспонируем А, чтобы получить $B = A^T$ ($0 \leq i, j < n, b[i][j] \in B : b[i][j] = a[j][i]$).
   Шаг 2. Если $A = B$ ($b[i][j] \in B : b[i][j] = a[i][j]$, где $0 \leq i, j < n$), то бинарное отношение будет является симметричным, иначе отношение не симметрично.
   Асимптотика $O(n^2)$.
3. Алгоритм 3 - Проверка бинарного отношения на транзитивность:
   Бинарное отношение транзитивным тогда и только тогда, когда $\rho\rho \subset \rho$. Это означает, что бинарное отношение ρ транзитивно, если $M(\rho)M(\rho) \leq M(\rho)$.
   Вход: матрица бинарного отношения $A = (a_{ij})$, размерности $n \times n$
   Выход: "Множество транзитивно"или "Множество не транзитивно"

На вход подается матрица бинарного отношения А.

Шаг 1. Получить матрицу $B = A^2$.

Шаг 2. Сравнить полученную и исходную матрицу.

Шаг 3. Если $B \leq A$ ($b[i][j] \in B : b[i][j] \leq a[i][j]$, где $0 \leq i, j < n$), то бинарное отношение транзитивно, иначе не транзитивным.

Асимптотика $O(n^3)$.

4. Алгоритм 4 - Проверка бинарного отношения на антирефлексивность:

   Вход: матрица бинарного отношения $A = (a_{ij})$, размерности $n \times n$

   Выход: "Множество антирефлексивно"или "Множество не антирефлексивно"

   На вход подается матрица бинарного отоношения А.

   Шаг 1. Суммирование элементов на главной диагонали ($sum = \sum\limits_{i=1}^{n} a[i][i]$).

   Шаг 2. Если $sum = 0$, то отношение является антирефлексивным, иначе не антирефлексивным.

   Асимптотика $O(n)$.

5. Алгоритм 5 - Проверка бинарного отношения на антисимметричность:

   Вход: матрица бинарного отношения $A = (a_{ij})$, размерности $n \times n$

   Выход: "Множество антисимметрично"или "Множество не антисимметрично"

   Шаг 1. Транспонируем А, чтобы получить $C = A^T$.

   Шаг 2. Получим матрицу B поэлементным умножением матрицы A на C.

   Шаг 4. Если $b[i][j] = 0$, где $b[i][j]$ элемент матрицы B, $0 \leq i, j < n$ и $i \neq j$, то отношение является антисиметричным, иначе отношение не антисиметрично.

   Асимптотика $O(n^2)$.

6. Алгоитм 6 - Классификация бинарного отношения:

   Вход: матрица бинарного отношения $A = (a_{ij})$, размерности $n \times n$

   Выход: «Бинарное отношение является отношением квазипорядка», «Бинарное отношение является отношением эквивалентности», «Бинарное отношение является отношением частичного порядка» или «Бинарное отношение является отношением строгого порядка».

   Шаг 1. Запустить алгоритмы 1 и 3 (проверки на рефлексивность и транзитивность), подав им на вход матрицу А. Если алгоритмы вернут значения «Бинарное отношение является рефлексивным» и «Бинарное отношение является транзитивным», то вернуть значение «Бинарное отношение является отношением квазипорядка».

Шаг 2. Запустить алгоритмы 1, 2 и 3 (проверки на рефлексивность, симметричность и транзитивность), подав им на вход матрицу А. Если алгоритмы вернут значения «Бинарное отношение является рефлексивным», «Бинарное отношение является симметричным» и «Бинарное отношение является транзитивным», то вернуть значение «Бинарное отношение является отношением эквивалентности».

Шаг 3. Запустить алгоритмы 3 и 5 (проверки на антисимметричность и транзитивность), подав им на вход матрицу А. Если алгоритмы вернут значения «Бинарное отношение является антисимметричным» и «Бинарное отношение является транзитивным», то вернуть значение «Бинарное отношение является отношением частичного порядка».

Шаг 4. Запустить алгоритмы 3, 4 и 5 (проверки на антирефлексивность, антисимметричность и транзитивность), подав им на вход матрицу А. Если алгоритмы вернут значения «Бинарное отношение является антирефлексивным», «Бинарное отношение является антисимметричным» и Бинарное отношение является транзитивным», то вернуть значение «Бинарное отношение является отношением строгого порядка».

Если не учитывать сложность вызываемых алгоритмов, то асимптотика $O(1)$, иначе асимптотика $O(n^3)$.

3.2 Описание алгоритмов построения основных замыканий бинарных отношений

1. Алгоритм 7 - Замыкание бинарного отношения относительно рефлексивности:

   Вход: матрица бинарного отношения $A = (a_{ij})$, размерности $n \times n$

   Выход: матрица бинарного отношения A', замкнутая относительно рефлексивности

   Шаг 1. Присвоить каждому $a[i][i]$ значение 1, где $0 \leq i < n :$, после чего вернуть полученную матрицу бинарного отношения $A' = (a'_{ij})$ с построенным на нем рефлексивным замыканием.

   Асимптотика $O(n)$.

2. Алгоритм 8 - Замыкание бинарного отношения относительно симметричности:

Вход: матрица бинарного отношения $A = (a_{ij})$, размерности $n \times n$

Выход: матрица бинарного отношения A', замкнутая относительно симметричности

Шаг1. Каждому элементу $a[i][j]$ $0 \leq i, j < n$ матрицы А присваивается значение элемента $a[j][i]$, после чего вернуть полученную матрицу бинарного отношения $A' = (a'_{ij})$ с построенным на нем симметричным замыканием.

Асимптотика $O(n^2)$.

3. Алгоритм 9 - Замыкание бинарного отношения относительно транзитивности:

   Вход: матрица бинарного отношения $A = (a_{ij})$, размерности $n \times n$

   Выход: матрица бинарного отношения A', замкнутая относительно транзитивности

   Шаг1. Если $a[i][k] = 1$ и $a[k][j] = 1$, то присвоить $a[i][j]$ значение 1, где $0 \leq i, j, k < k$. Такой шаг нужно повторить n раз в силу определения п.3) оператора транзитивного замыкания в лемме 2, после чего вернуть полученную матрицу бинарного отношения $A' = (a'_{ij})$ с построенным на нем транзитивным замыканием.

   Асимптотика $O(n^4)$.

3.3 Коды программ, реализующей рассмотренные алгоритмы

3.4 Результаты тестирования программ

```
How you want enter your relation (set or matrix)?
set
Enter the number of elements in relation
3
Enter your set
(1,2) (1,3)
----------------------------------------
Relation's matrix
[[0 1 1]
 [0 0 0]
 [0 0 0]]
----------------------------------------
Set is anti-reflexive
Set is anti-symmetry
Set is transitive
----------------------------------------
Relation is the relation of the strict order
----------------------------------------
Closure matrix:
[[1 1 1]
 [1 1 0]
 [1 0 1]]
----------------------------------------
Closure set:
(1, 1) (1, 2) (1, 3) (2, 1) (2, 2) (3, 1) (3, 3)
----------------------------------------
```

Рисунок 1 – Ввод бинарного отношения (1, 2), (1, 3) в виде множества с выводом свойств этого множества и замыкания относительно эквивалентности

```
How you want enter your relation (set or matrix)?
set
Enter the number of elements in relation
5
Enter your set
(3,2) (2,5) (4,2)
----------------------------------------
Relation's matrix
[[0 0 0 0 0]
 [0 0 0 0 1]
 [0 1 0 0 0]
 [0 1 0 0 0]
 [0 0 0 0 0]]
----------------------------------------
Set is anti-reflexive
Set is not symmetry
Set is anti-transitive
----------------------------------------
Closure matrix:
[[1 0 0 0 0]
 [0 1 1 1 1]
 [0 1 1 1 1]
 [0 1 1 1 1]
 [0 1 1 1 1]]
----------------------------------------
Closure set:
(1, 1) (2, 2) (2, 3) (2, 4) (2, 5) (3, 2) (3, 3) (3, 4) (3, 5) (4, 2) (4, 3) (4, 4) (4, 5) (5, 2) (5, 3) (5, 4) (5, 5)
----------------------------------------
```

Рисунок 2 – Ввод бинарного отношения (3, 2), (2, 5), (4, 2) в виде множества с выводом свойств этого множества и замыкания относительно эквивалентности

```
How you want enter your relation (set or matrix)?
matrix
Enter the number of elements in relation
5
Enter your matrix
0 0 0 0 0
0 0 0 0 1
0 1 0 0 0
0 1 0 0 0
0 0 0 0 0
----------------------------------------
Relation's set
(2, 5) (3, 2) (4, 2)
----------------------------------------
Set is anti-reflexive
Set is not symmetry
Set is anti-transitive
----------------------------------------
Closure matrix:
[[1 0 0 0 0]
 [0 1 1 1 1]
 [0 1 1 1 1]
 [0 1 1 1 1]
 [0 1 1 1 1]]
----------------------------------------
Closure set:
(1, 1) (2, 2) (2, 3) (2, 4) (2, 5) (3, 2) (3, 3) (3, 4) (3, 5) (4, 2) (4, 3) (4, 4) (4, 5) (5, 2) (5, 3) (5, 4) (5, 5)
----------------------------------------
```

Рисунок 3 – Ввод бинарного отношения (3, 2), (2, 5), (4, 2) в виде матрицы с выводом свойств этого множества и замыкания относительно эквивалентности

```
How you want enter your relation (set or matrix)?
matrix
Enter the number of elements in relation
5
Enter your matrix
0 1 0 0 0
0 0 1 0 0
0 0 0 0 0
0 0 1 0 0
0 0 0 0 0
----------------------------------------
Relation's set
(1, 2) (2, 3) (4, 3)
----------------------------------------
Set is anti-reflexive
Set is not symmetry
Set is anti-transitive
----------------------------------------
Closure matrix:
[[1 1 1 1 0]
 [1 1 1 1 0]
 [1 1 1 1 0]
 [1 1 1 1 0]
 [0 0 0 0 1]]
----------------------------------------
Closure set:
(1, 1) (1, 2) (1, 3) (1, 4) (2, 1) (2, 2) (2, 3) (2, 4) (3, 1) (3, 2) (3, 3) (3, 4) (4, 1) (4, 2) (4, 3) (4, 4) (5, 5)
----------------------------------------
```

Рисунок 4 – Ввод бинарного отношения (1, 2), (2, 3), (4, 3) в виде матрицы с выводом свойств этого множества и замыкания относительно эквивалентности

3.5 Оценки сложности рассмотренных алгоритмов

3.5.1 Алгоритм определения рефлексивности

Сложность выполнения проверки на рефлексивность определяется как $O(n)$.

3.5.2 Алгоритм определения антирефлексивности

За счет схожести с алгоритмом определения рефлексивности сложность определяется как $O(n)$.

3.5.3 Алгоритм определения симметричности

Сложность транспонирования в numpy определяется как $O(n^{3/2} log\, n)$, сложность сравнение двух матриц поэлементно определяется как $O(n^2)$. Отсюда можно сделать вывод, что сложность алгоритма будет определятся как $O(n^{3/2} log\, n + n^2) = O(n^2)$.

3.5.4 Алгоритм определения антисимметричности

Сложность поэлементного умножения матриц определяется как $O(n^2)$, сложность сравнение двух матриц поэлементно определяется как $O(n^2)$. Отсюда можно сделать вывод, что сложность будет определятся как $O(n^2+n^2) = O(n^2)$

3.5.5 Алгоритм определения транзитивности

Из всего выше сказанного очевидно, что сложность проверки на транзитивность или антитранзитивность составляет $O(n^3)$, так как в нем используется умножение, сравнение матриц и тройной цикл для проверки рефлексивности в худшем случае матриц.

3.5.6 Алгоритм классификации

Сложность выполнения самого алгоритма классификации бинарных отношений реализованно через словарь языка python и оператор if, поэтому является константной ($O(1)$), если не учитывать сложность выполнения проверки свойств отношения. Если учитывать сложность алгоритмов проверки свойств отношения, то :

1. Сложность проверка на квазипорядок определяется как $O(n^3 + n) = O(n^3)$.
2. Сложность проверка на эквивалентность определяется как $O(n^3 + n + n^{3/2}log\ n) = O(n^3)$.
3. Сложность проверка на частичный порядок определяется как $O(n^3 + n + n^3) = O(n^3)$.
4. Сложность проверка на строгий порядок определяется как $O(n^3 + n + n^3) = O(n^3)$.

3.5.7 Построение замыкания рефлексивности

Так как весь алгоритм строится на заполнении главной диагонали матрицы 1, то его сложность состовляет $O(n)$.

3.5.8 Построение замыкания симметричности

Для посторения замыкания симметричности используются вложенный цикла, поэтому сложность алгоритма определяется как $O(n^2)$.

3.5.9 Построение замыкания транзитивности

Для посторения замыкания транзитивности используются три вложенный цикла, поэтому сложность алгоритма определяется как $O(n^4)$.

ЗАКЛЮЧЕНИЕ

В рамках данной лабораторной работы были рассмотренны теоритические основы свойств бинарных отношений, их видов и методов их замыкания по каждому из свойств. На основе этой теоретической части была смоделирована программа на языке Python с использованием средств библиотеки Numpy, которая способна определить свойства заданного множества, его вид и построить систему замыкания по каждому из основных свойств бинарного отношения, а так же была оценена асимптотика каждого реализованного алгоритма.