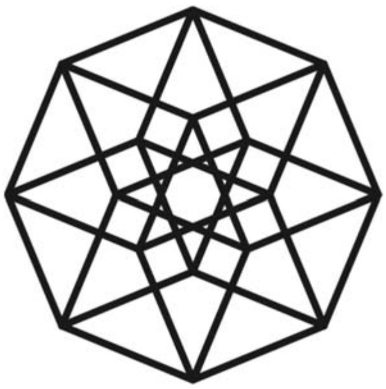
МЕЖГОСУДАРСТВЕННОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ОБРАЗОВАНИЯ
«БЕЛОРУССКО-РОССИЙСКИЙ УНИВЕРСИТЕТ»

Кафедра «Высшая математика»

ДИСКРЕТНАЯ МАТЕМАТИКА И МАТЕМАТИЧЕСКОЕ МОДЕЛИРОВАНИЕ

Методические рекомендации к лабораторным работам для студентов специальностей 6-05-0713-04 «Автоматизация технологических процессов и производств» и 6-05-0714-02 «Технология машиностроения, металлорежущие станки и инструменты» дневной и заочной форм обучения



Могилев 2025

УДК 519 (075.8)
ББК 22.176
Д48

Рекомендовано к изданию
учебно-методическим отделом
Белорусско-Российского университета

Одобрено кафедрой «Высшая математика» «28» ноября 2024 г., протокол № 3

Составитель ст. преподаватель А. Г. Козлов

Рецензент канд. техн. наук М. Н. Миронова

Методические рекомендации к лабораторным работам содержат необходимые теоретические сведения по курсу «Дискретная математика и математическое моделирование», примеры с решениями и задания для самостоятельной работы.

Учебное издание

ДИСКРЕТНАЯ МАТЕМАТИКА
И МАТЕМАТИЧЕСКОЕ МОДЕЛИРОВАНИЕ

Ответственный за выпуск	В. Г. Замураев
Корректор	И. В. Голубцова
Компьютерная верстка	Н. П. Полевничая

Подписано в печать . Формат 60×84/16. Бумага офсетная. Гарнитура Таймс.
Печать трафаретная. Усл. печ. л. . Уч.-изд. л. . Тираж 56 экз. Заказ №

Издатель и полиграфическое исполнение:
Межгосударственное образовательное учреждение высшего образования
«Белорусско-Российский университет».
Свидетельство о государственной регистрации издателя,
изготовителя, распространителя печатных изданий
№ 1/156 от 07.03.2019.
Пр-т Мира, 43, 212022, г. Могилев.

Содержание

1 Лабораторная работа № 1. Множества

Цель работы: изучение свойств множеств, операций над множествами и тождественных преобразований над множествами.

Теоретические сведения

Основное понятие: *множество*. Множества будем обозначать прописными латинскими буквами: A, B, …, а их элементы – малыми: a, b, ...

Подмножеством A множества S $\left(A \subseteq S\right)$ называется любое множество, все элементы которого принадлежат S, т. е. $A \subseteq S$, если $a \in A \Rightarrow a \in S$.

Множества A и B *совпадают* или *равны* (A=B), если они состоят из одних и тех же элементов. Иначе говоря, $A = B \Leftrightarrow A \subseteq B$ и $B \subseteq A$.

Если $A \subseteq B$, но $A \neq B$, то говорят, что A – *собственное подмножество* множества B, и записывают $A \subset B$.

Пустое множество $\varnothing$ – множество без элементов. Очевидно, любое множество содержит в качестве подмножества пустое множество.

Если каждому элементу из множества A поставлен в соответствие некоторый элемент из множества B, то говорят, что задано *отображение* φ из множества A в множество B. Обозначение: $\varphi : A \to B$. Если $b = \varphi\left(a\right)$, $a \in A$ и $b \in B$, то b называют образом элемента a при отображении φ, а a – прообразом элемента b.

Два отображения $\varphi_1 : A_1 \to B_1$ и $\varphi_2 : A_2 \to B_2$ называются *равными*, если $A_1 = A_2$, $B_2 = B_2$ и $\varphi_1\left(a\right) = \varphi_2\left(a\right)$, $\forall a \in A$.

Отображение $\varphi : A \to B$ называется *вложением*, если каждый элемент из B имеет не более одного прообраза, т. е. $\varphi\left(a\right) = \varphi\left(b\right) \Rightarrow a = b$. Если каждый элемент из B имеет хотя бы один прообраз, то отображение φ называется *наложением*. Отображение, являющееся одновременно включением и наложением, называется *взаимно-однозначным*.

Если $A = B$, то взаимно-однозначное отображение $\varphi : A \to A$ называется *подстановкой.*

Множества могут быть *конечными* (т. е. состоящими из конечного числа элементов) и *бесконечными.* Число элементов в множестве A называется *мощностью* множества A и обозначается $\left|A\right|$. Множества A и B *равномощны,* если существует взаимно-однозначное отображение $\varphi : A \to B$.

Конечные множества A такие, что $\left|A\right| = n$, будем называть *n-множествами.*

Бесконечное множество называется *счетным*, если оно равномощно множеству натуральных чисел. В дальнейшем будем рассматривать только счетные бесконечные множества.

Операции на множестве.

1 *Объединение* (или *сумма*) $A \cup B$ множеств A и B – множество всех элементов, принадлежащих либо A, либо B, либо A и B одновременно: $A \cup B = \{a \in S \,|\, a \in A \text{ или } a \in B\}$.

2 *Пересечение* (*произведение*) множеств A и B: $A \cap B = \{a \in S \,|\, a \in A \text{ и } a \in B\}$.

3 *Дополнение* множества A: $\overline{A} = \{a \in S \,|\, a \notin A\}$.

4 *Разность* множеств A и B: $A \setminus B = \{a \in S \,|\, a \in A \text{ и } a \notin B\}$.

5 *Симметрическая разность* множеств A и B: $A \Delta B = (A \setminus B) \cup (B \setminus A)$.

Постановка задачи

Задание

Сформировать множества A, B и C из целых произвольным образом, взяв от десяти до двадцати элементов. Проверить выполнимость тождеств, заданных таблицей 1.1.

Таблица 1.1 – Варианты заданий

Вариант	Тождество	Вариант	Тождество
1	$A \setminus (B \cup C) = (A \setminus B) \cap (A \setminus C)$	16	$(A \cup (A\Delta B) \cup (A\Delta C)) \setminus ((B \cup C) \cap \overline{A}) = A$
2	$A \cap (B \setminus C) = (A \cap B) \setminus (A \cap C)$	17	$(A \setminus B) \cap (A \setminus C) = (A\Delta (B \cup C)) \setminus (B \cup C)$
3	$(A \setminus B) \setminus C = (A \setminus C) \setminus (B \setminus C)$	18	$(A \cap B \cap \overline{C}) \Delta (A \cap B \cap C) = A \cap B$
4	$A \cap (B \setminus C) = (A \cap B) \setminus C$	19	$(A \setminus B) \Delta (A \setminus C) = A \cap \overline{B} \cap C \cup A \cap B \cap \overline{C}$
5	$A \setminus (B \setminus C) = (A \setminus B) \cup (A \cap C)$	20	$(A \cup B) \Delta (A \cup C) = \overline{A} \cap (B \cap \overline{C} \cup \overline{B} \cap C)$
6	$A \setminus (A \setminus B) = A \cap B$	21	$(A\Delta B) \setminus (A \cup C) = B \cap \overline{A} \cap \overline{C}$
7	$A \cup (B \setminus C) = (A \cup B) \cap (A \setminus \overline{C})$	22	$(A \cap B) \cup (A \cap \overline{B}) = (A \cup B) \cap (A \cup \overline{B})$
8	$A \setminus (B \cup C) = (A \setminus B) \setminus C$	23	$(A \setminus B) \Delta (B \setminus C) = (A \cap \overline{B}) \cup (\overline{A} \cap B \cap \overline{C})$
9	$A \setminus (B \cap C) = (A \setminus B) \cup (A \setminus C)$	24	$(A \setminus B \setminus C) \Delta (B \cup C) = A \cap \overline{B} \cap \overline{C} \cup B \cap C$
10	$(A \cap \overline{B}) \setminus (A \cap C) = (A \setminus B) \setminus C$	25	$((A\Delta B) \cup (A\Delta C)) \setminus (B \cup C) = (A \setminus B) \setminus C$
11	$A \cup B = A\Delta B\Delta (A \cap B)$	26	$(A\Delta B) \cap (B\Delta C) = A \cap \overline{B} \cap C \cup \overline{A} \cap B \cap \overline{C}$
12	$(A \cup B) \Delta (A \cap B) = A\Delta B$	27	$(A \cup B) \Delta (A \setminus B) = B$
13	$(A \setminus B) \Delta (A \cap B) = A$	28	$((A \cap B) \setminus A) \cup (A\Delta B) = (C \cup B) \setminus (C \cap B)$
14	$(A\Delta B) \Delta (B\Delta C) = A\Delta C$	29	$(A \cap B) \Delta (A \cup B) = (A \cup B) \cup (A\Delta B)$
15	$A \cup B = (A\Delta B) \cup (A \cap B)$	30	$(A \cap \overline{B} \cap C) \cup (\overline{A} \cap \overline{B} \cap C) \cup (B \cap C) = C$

Ход выполнения работы

Отчеты по лабораторным работам формируются в Google Colaboratory (Golab), при этом используются LaTex и язык текстовой разметки Markdown. Пример отчета приведен на рисунке 1.1.

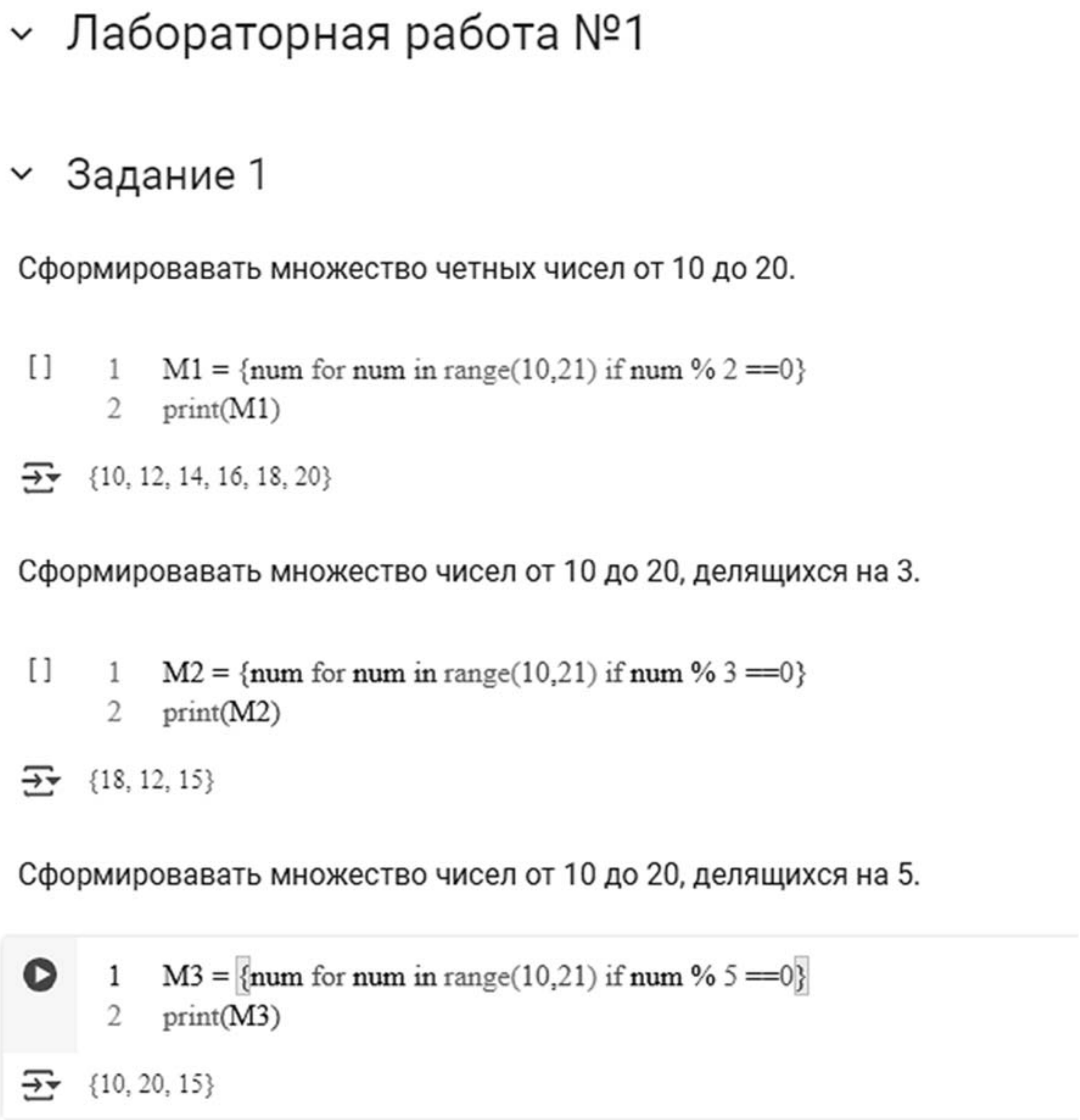


Рисунок 1.1 – Фрагмент отчета по лабораторной работе

Доступ к отчетам предоставляется преподавателю посредствам функции «Поделиться», реализованной в Google Диске.

Контрольные вопросы

1 Сформируйте понятие множества, приведете примеры множеств.
2 Дайте определение операции: пересечение множеств.
3 Дайте определение операции: объединение множеств.
4 Дайте определение операции: разность множеств.

5 Дайте определение операции: симметричная разность множеств.

6 Дайте определение подмножества, приведите примеры.

2 Лабораторная работа № 2. Отношения, функции

Цель работы: изучение бинарных отношений и функциональных зависимостей; определение основных свойств бинарных отношений, способов задания и графического изображения.

Теоретические сведения

Если элементы двух множеств X и Y различной природы сопоставить между собой по какому-либо правилу, т. е. для каждого $x \in X$ указать один или несколько элементов множества Y, то может быть сформировано множество пар $(x;\ y)$, являющееся подмножеством прямого произведения множеств X и Y, т. е.

$$\{(x;\, y) \mid x \in X,\, y \in Y\} \subseteq (X \times Y).$$

Множество X чаще всего называют областью отправления, а множество Y – областью прибытия. Значение $y \in Y$ называют образом для конкретного значения $x_i \in X$, а значение $x \in X$ – прообразом для конкретного значения $y_j \in Y$.

Свойства отношений.

Бинарное отношение *рефлексивно*, если для любого x_i имеем $r(x_i;\ x_i) = 1$, т. е. отношение имеет значение «истины» при применении к одному элементу x_i; такими отношениями являются «быть равным», «быть похожим», «быть изоморфным», «быть эквивалентным» и т. п. При матричном задании такого отношения это означает, что на главной диагонали матрицы находятся только «1», а при графическом представлении – петли при каждой вершине графа.

Бинарное отношение *антирефлексивно*, если для любого x_i имеем $r(x_i;\ x_i) = 0$, т. е. отношение имеет значение «ложь» применительно к одному элементу x_i. Такими отношениями являются «быть больше», «быть меньше», «быть родителем» и т. п. При матричном задании такого отношения это означает, что на главной диагонали матрицы находятся только «0», а при графическом представлении – отсутствие петель при каждой вершине графа.

Бинарное отношение *симметрично*, если для любой пары $(x_i;\ x_j)$ имеем $r(x_i;\ x_j) = r(x_j\ x_i) = 1$. Это могут быть такие отношения как «быть похожим», «быть эквивалентным», «быть родственником» и т. п. При матричном задании такого отношения это означает симметричное расположение «1» относительно главной диагонали, при графическом представлении – отсутствие стрелок на линиях, соединяющих вершины x_i и x_j, или их наличие, но в обе стороны.

Бинарное отношение *антисимметрично*, если для любой пары $(x_i;\ x_j)$ при $i \neq j$ имеем $r(x_i;\ x_j) \neq r(x_j;\ x_i)$, а при $i = j$ $r(x_i;\ x_i) = 1$. Такими отношениями являются «быть больше или равным», «быть меньше или равным» и т. п. При матричном задании такого отношения это означает несимметричное расположение «1» относительно главной диагонали, но наличие их на главной диагонали, при графическом представлении – наличие стрелок на линиях, соединяющих вершины x_i и x_j и наличие петель у вершин графа.

Бинарное отношение *асимметрично*, если для любой пары (x_i; x_j) имеем $r(x_i; x_j) \neq r(x_j; x_i)$. Такими отношениями являются «быть больше», «быть меньше», «быть родителем» и т. п. При матричном задании такого отношения это означает только несимметричное расположение «1» относительно главной диагонали и наличие только «0» на ней, а при графическом представлении наличие стрелок на линиях, соединяющих вершины x_i и x_j, и отсутствие петель у вершин графа.

Бинарное отношение *транзитивно*, если для любых трех элементов x_i, x_j, x_k имеем $r(x_i; x_j) = 1$ только при условии $r(x_i; x_k) = 1$ и $r(x_k; x_j) = 1$. Такими отношениями являются «быть больше», «быть меньше», «быть родственником» и т. п. При матричном представлении это означает, что если $r(x_i; x_k) = 1$ и $r(x_k; x_j) = 1$, то это же отношение можно установить между вершинами x_i и x_j через промежуточную вершину x_k, т. е. найти $r(x_i; x_j) = 1$; при графическом представлении – наличие пути из вершины x_i в вершину x_j через промежуточную вершину x_k, используя ребра (x_i; x_k) и (x_k; x_j).

Типы отношений.

Отношение эквиваленции. Бинарное отношение $R \subseteq (X \times X)$, удовлетворяющее условиям рефлексивности, симметричности, транзитивности, называют отношением эквиваленции. Такими отношениями могут быть «быть равным», «быть похожим», «быть одинаковым», «быть родственником» и т. п. Отношение эквиваленции принято обозначать знаком $r_{\sim}(x_i; x_j)$ или $\sim(x_i; x_j)$. Используя отношение эквиваленции, можно формировать классы эквиваленции $K(x_\alpha)$ по заданному образцу x_α в виде подмножеств X_α множества X, т. е. $K(x_\alpha) = X_\alpha = \{x_i \mid r_{\sim}(x_i; x_\alpha) = 1, x_i, x_\alpha \in X\} \subseteq X$.

Отношение порядка. Бинарные отношения $R \subseteq (X \times X)$, удовлетворяющие условиям рефлексивности, антисимметричности и транзитивности называют, отношением порядка. Такими отношениями являются «быть не больше», «быть не меньше», «быть не старше» и т. п. Отношение порядка принято обозначать для элементов множества $r \leq (x_i; x_j)$ или $\leq (x_i; x_j)$, а для множеств – $r \subseteq (x_i; x_j)$ или $\subseteq (x_i; x_j)$. Использование отношения порядка на одном множестве X позволяет упорядочить элементы этого множества, т. е., рассматривая отношение на каждой паре элементов множества, устанавливать частичный порядок на всём множестве X. Примерами частично упорядоченных множеств являются множество целых чисел с заданным отношением порядка, т. е. {1; 2; 3; ...}, множество действительных чисел, в том числе положительных и отрицательных; счетные множества нематематических объектов, упорядоченные по значениям индексов, т. е. X_1, X_2, ...; счетные множества букв и символов, упорядоченные алфавитом; множество подмножеств универсального множества с отношением включения $\subseteq (x_i, x_j)$ и т. п.

Отношение строгого порядка. Бинарное отношение $R \subseteq (X \otimes X)$, удовлетворяющее условиям антирефлексивности, асимметричности и транзитивности называют, отношением строгого порядка. Такими отношениями могут быть: «быть больше», «быть меньше», «быть частью», «быть подчиненным» и т. п. Использование отношения строго порядка формирует линейную упорядоченность элементов множества X. Для обозначения отношения строгого порядка приняты символы: между элементами множества – $r < (x_i; x_j)$ или $< (x_i; x_j)$, между множествами – $r \subset (x_i; x_j)$ или $\subset (x_i; x_j)$.

Постановка задачи

Выполнить следующие упражнения.

Задание 1

Пусть M – некоторое множество; S и R – некоторые бинарные отношения, причем $R \subseteq M \times M$. Задать списком отношение R, обратное отношение R^{-1}, дополнение $\overline{R}$. Изобразить графически R и записать его матрицу. Найти $R \cap S$, $R \setminus S$, $R \circ S$, $S \circ R$, если:

1) $M = \{1, 3, 5, 7\}$, $R = \{(a, b) \mid a \le b\}$, $S = \{(3, 1), (1, 5), (5, 3), (5, 7)\}$;
2) $M = \{1, 3, 5, 7\}$, $R = \{(a, b) \mid a + 2 = b\}$, $S = \{(7, 5), (5, 3), (3, 5), (7, 1)\}$;
3) $M = \{1, 3, 5, 7\}$, $R = \{(a, b) \mid (a + b) / 2 \in M\}$, $S = \{(1, 5), (1, 7), (1, 3)\}$;
4) $M = \{1, 3, 5, 7\}$, $R = \{(a, b) \mid a + b - 1 \in M\}$, $S = \{(1, 3), (1, 7), (3, 5)\}$;
5) $M = \{1, 3, 5, 7\}$, $R = \{(a, b) \mid a - 2 = \mathrm{b}\}$, $S = \{(1, 3), (7, 5), (1, 7), (5, 3)\}$;
6) $M = \{1, 3, 5, 7\}$, $R = \{(a, b) \mid 2a + b \in M\}$, $S = \{(1, 3), (3, 1), (1, 5), (1, 7)\}$;
7) $M = \{2, 4, 6, 8\}$, $R = \{(a, b) \mid b - a \in M\}$, $S = \{(8, 6), (6, 8), (2, 6), (6, 4)\}$;
8) $M = \{2, 4, 6, 8\}$, $R = \{(a, b) \mid b - 2 = a\}$, $S = \{(6, 4), (4, 6), (2, 8), (4, 4)\}$;
9) $M = \{2, 4, 6, 8\}$, $R = \{(a, b) \mid 1+(a / b) \in M\}$, $S = \{(8, 8), (2, 4), (8, 2)\}$;
10) $M = \{2, 4, 6, 8\}$, $R = \{(a, b) \mid 2b - a \in M\}$, $S = \{(6, 6), (6, 4), (4, 6), (2, 8)\}$;
11) $M = \{2, 4, 6, 8\}$, $R = \{(a, b) \mid b + a \in M\}$, $S = \{(2, 2), (4, 2), (6, 2), (8, 8)\}$;
12) $M = \{2, 4, 6, 8\}$, $R = \{(a, b) \mid ab \le 24\}$, $S = \{(4, 4), (6, 4), (8, 4), (8, 6)\}$;
13) $M = \{2, 4, 6, 8\}$, $R = \{(a, b) \mid a + b - 2 \in M\}$, $S = \{(2, 8), (4, 6), (8, 6), (4, 4)\}$;
14) $M = \{0, 1, 2, 3, 4, 5\}$, $R = \{(a, b) \mid b - a < 2\}$, $S = \{(3, 1), (1, 3), (4, 5), (3, 5)\}$;
15) $M = \{0, 1, 2, 3, 4\}$, $R = \{(a, b) \mid \sin(0{,}5\pi[a+b]) \ne 0\}$, $S = \{(0, 1), (1, 1), (3, 4)\}$;
16) $M = \{0, 1, 2, 3, 4\}$, $R = \{(a, b) \mid \cos(0{,}5\pi b) = 1\}$, $S = \{(0, 3), (2, 4), (4, 0)\}$;
17) $M = \{0, 1, 2, 3, 4\}$, $R = \{(a, b) \mid (-1)^a + (-1)^b \le 0\}$, $S = \{(0, 3), (2, 2), (4, 2)\}$;
18) $M = \{1, 2, 3, 4, 5\}$, $R = \{(a, b) \mid a + b$ – четное$\}$, $S = \{(1, 3), (5, 1), (3, 5)\}$;
19) $M = \{1, 2, 3, 4, 5\}$, $R = \{(a, b) \mid a + b$ – нечетное$\}$, $S = \{(2, 2), (3, 4), (4, 5)\}$;
20) $M = \{у, д, г, о, е, ж\}$, $R = \{(a, b) \mid a$ и b –гласные$\}$, $S = \{(г, о), (о, е), (е, у)\}$;
21) $M = \{у, д, г, о, е, м\}$, $R = \{(a, b) \mid a$ – согласная$\}$, $S = \{(г, д), (д, г), (у, у)\}$;
22) $M = \{г, о, е, ж, к, м\}$, $R = \{(a, b) \mid b$ – гласная$\}$, $S = \{(д, ж), (ж, о), (о, ж)\}$;
23) $M = \{д, о, е, к\}$, $R = \{(a, b) \mid a$ – гласная, b – согласная$\}$, $S = \{(к, д), (к, о)\}$;
24) $M = \{д, г, о, е\}$, $R = \{(a, b) \mid a$ – согласная, b – гласная$\}$, $S = \{(г, д), (м, м)\}$;
25) $M = \{2, -3, 4, -5\}$, $R = \{(a, b) \mid a + b < 0\}$, $S = \{(-5, 1), (-3, 4), (-3, -5)\}$;

26) $M = \{-1, 2, -3, 4\}$, $R = \{(a, b) \mid ab > 0\}$, $S = \{(-1, 3), (4, 2), (-3, -1), (2, 4)\}$;
27) $M = \{-1, 2, -3, -5\}$, $R = \{(a, b) \mid a + \mathrm{b} > 0\}$, $S = \{(-1, 2), (-3, 2), (-5, -5)\}$;
28) $M = \{-1, -3, 4, -5\}$, $R = \{(a, b) \mid ab < 0\}$, $S = \{(-3, -3), (-3, -5), (-1, 4)\}$;
29) $M = \{-1, 2, 4, -5\}$, $R = \{(a, b) \mid b / a \in Z\}$, $S = \{(-5, -1), (-1, -5), (2, 4)\}$;
30) $M = \{-1, 4, -5\}$, $R = \{(a, b) \mid a / b \in N\}$, $S = \{(-5, -5), (-1, -1), (-1, 4)\}$.

Задание 2

Выяснить свойства отношения, заданного матрицей:

1) $\begin{pmatrix} 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 1 \end{pmatrix}$; 2) $\begin{pmatrix} 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix}$; 3) $\begin{pmatrix} 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 \end{pmatrix}$; 4) $\begin{pmatrix} 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 \end{pmatrix}$;

5) $\begin{pmatrix} 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 \end{pmatrix}$; 6) $\begin{pmatrix} 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 \end{pmatrix}$; 7) $\begin{pmatrix} 1 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 \end{pmatrix}$; 8) $\begin{pmatrix} 0 & 1 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 & 1 \\ 1 & 1 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 1 & 0 \end{pmatrix}$;

9) $\begin{pmatrix} 0 & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 \end{pmatrix}$; 10) $\begin{pmatrix} 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 \end{pmatrix}$; 11) $\begin{pmatrix} 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 1 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 & 0 \\ 1 & 1 & 0 & 0 & 0 \end{pmatrix}$; 12) $\begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 \end{pmatrix}$;

13) $\begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 \end{pmatrix}$; 14) $\begin{pmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \end{pmatrix}$; 15) $\begin{pmatrix} 1 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 \end{pmatrix}$; 16) $\begin{pmatrix} 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \end{pmatrix}$;

17) $\begin{pmatrix} 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 \end{pmatrix}$; 18) $\begin{pmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{pmatrix}$; 19) $\begin{pmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix}$; 20) $\begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix}$;

21) $\begin{pmatrix} 1 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 1 \end{pmatrix}$; 22) $\begin{pmatrix} 1 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix}$; 23) $\begin{pmatrix} 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \end{pmatrix}$; 24) $\begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 \end{pmatrix}$;

25) $\begin{pmatrix} 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 \\ 1 & 0 & 0 & 1 \end{pmatrix}$; 26) $\begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 \end{pmatrix}$; 27) $\begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 0 \end{pmatrix}$; 28) $\begin{pmatrix} 1 & 0 & 1 \\ 0 & 0 & 0 \\ 1 & 0 & 1 \end{pmatrix}$;

29) $\begin{pmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 1 \end{pmatrix}$; 30) $\begin{pmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{pmatrix}$; 31) $\begin{pmatrix} 1 & 0 & 1 \\ 0 & 0 & 0 \\ 1 & 0 & 1 \end{pmatrix}$; 32) $\begin{pmatrix} 1 & 0 & 1 \\ 0 & 0 & 0 \\ 1 & 0 & 1 \end{pmatrix}$.

Ход выполнения работы

1 Используя язык программирования Python (рисунок 2.1), построить множество M, реализовать S и R.

2 Задать списком отношение R, обратное отношение R^{-1}, дополнение $\overline{R}$.

3 Используя встроенные библиотеки Python, изобразить графически R.

4 Записать его матрицу R.

5 Найти $R \cap S$, $R \setminus S$, $R \circ S$, $S \circ R$ и записать их матрицы.

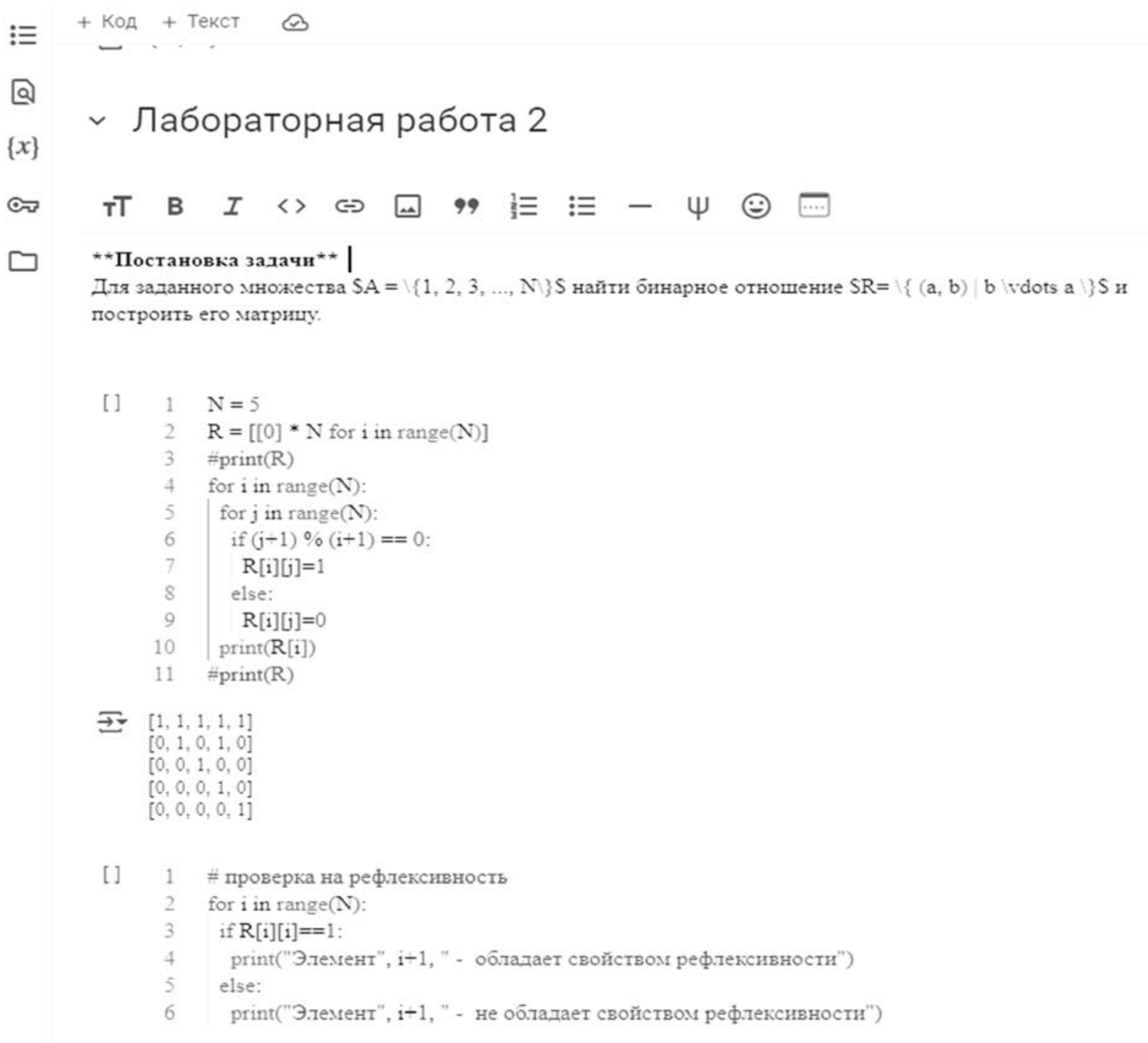


Рисунок 2.1 – Фрагмент отчета по лабораторной работе

Контрольные вопросы

1 Сформируйте понятие бинарного отношения, приведете примеры множеств.

2 Дайте определение рефлексивного бинарного отношения.

3 Дайте определение антирефлексивного бинарного отношения.

4 Дайте определение симметричного бинарного отношения.

5 Дайте определение асимметричного бинарного отношения.

6 Дайте определение антисимметричного бинарного отношения.
7 Дайте определение транзитивного бинарного отношения.
8 Дайте определение антитранзитивного бинарного отношения.

3 Лабораторная работа № 3. Булевы функции. Аналитическое представление булевых функций

Цель работы: изучение булевых функций, их способов задания; построение таблиц истинности.

Теоретические сведения

Булевой переменной x называется такая переменная, которая может принимать только два значения: ноль (0) или единица (1), т. е. $x \in \{0; 1\}$.

Функцию f, принимающую одно из двух значений, 0 или 1, от n переменных, каждая из которых принимает одно из двух значений, 0 или 1, будем называть *булевой функцией* $f(x_1, x_2, \ldots, x_n) = f\left(\tilde{x}^n\right)$ от n переменных.

Две булевы функции f_1 и f_2 называются *равными*, если они принимают одинаковые значения при всех наборах своих переменных.

Функция $f\left(x_1, ..., x_n\right)$ называется *существенно* зависящей от переменной x_i, если имеет место следующее равенство для какого-либо набора значений переменных $x_1, ..., x_{i-1}, x_{i+1}, ..., x_n$:

$$f(x_1, \ldots, x_{i-1}, 0, x_{i+1}, \ldots, x_n) \neq f(x_1, \ldots, x_{i-1}, 1, x_{i+1}, \ldots, x_n).$$

Если $f\left(x_1, ..., x_{i-1}, 0, x_{i+1}, ..., x_n\right) = f\left(x_1, ..., x_{i-1}, 1, x_{i+1}, ..., x_n\right)$ для любого набора значений переменных $x_1, ..., x_{i-1}, x_{i+1}, ..., x_n$, то x_i – *фиктивная переменная*.

Переменная x_i называется существенной, если при ее удалении изменяется значение функции, и несущественной, если значение функции не меняется.

Рассмотрим основные элементарные булевы функции (таблица 3.1).

Таблица 3.1 – Основные элементарные булевы функции

x	y	f_0	f_1	f_2	f_3	f_4	f_5	f_6	f_7	f_8	f_9	f_{10}	f_{11}	f_{12}	f_{13}	f_{14}	f_{15}
0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
0	1	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
1	0	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
1	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1

f_0 = const 0 – константа нуля;

$f_1 = xy$ – конъюнкция;

$f_8 = x \downarrow y$ – стрелка Пирса;

$f_9 = x \leftrightarrow y$ – эквивалентность;

$f_2 = x\bar{y}$ – запрет y; $f_{10} = \bar{y}$ – отрицание y;

$f_3 = x$ – повтор x; $f_{11} = x \leftarrow y$ – обратная импликация;

$f_4 = \bar{x}y$ – запрет x; $f_{12} = \bar{x}$ – отрицание x;

$f_5 = y$ – повтор y; $f_{13} = x \rightarrow y$ – прямая импликация;

$f_6 = x \oplus y$ – сложение по модулю 2; $f_{14} = x|y$ – штрих Шеффера;

$f_7 = x \vee y$ – дизъюнкция; $f_{15} = \text{const } 1$ – константа единицы.

Постановка задачи

Выполнить следующие упражнения.

Для формулы, заданной в таблице 3.2, построить таблицу истинности, по которой найти СДНФ.

Таблица 3 – Условия задания

Вариант	Формула	Вариант	Формула
1	$((x_1 \vee x_2) \leftrightarrow x_3) \rightarrow ((x_1 \wedge x_2) \vee x_3)$	16	$((x_1 \leftrightarrow x_2) \rightarrow x_3) \vee ((x_1 \rightarrow x_2) \vee x_3)$
2	$((x_1 \vee x_2) \wedge x_3) \leftrightarrow ((x_1 \wedge x_2) \vee \neg x_3)$	17	$(x_1 \rightarrow (x_1 \vee x_2)) \leftrightarrow (x_1 \vee x_2 \vee \neg x_3)$
3	$((x_1 \wedge x_2) \rightarrow x_3) \wedge ((x_1 \vee x_2) \rightarrow (x_2 \wedge x_3))$	18	$(\neg x_2 \rightarrow x_3) \wedge ((\neg x_1 \vee x_2) \rightarrow (x_2 \wedge \neg x_3))$
4	$((\neg x_1 \vee \neg x_2) \leftrightarrow (x_1 \wedge x_3)) \wedge (x_2 \vee x_3)$	19	$(x_3 \leftrightarrow (\neg x_1 \wedge \neg x_2)) \wedge (x_2 \rightarrow x_3)$
5	$(\neg x_1 \rightarrow (x_2 \leftrightarrow x_3)) \wedge (x_1 \vee x_2)$	20	$(x_1 \vee (x_2 \leftrightarrow x_3)) \rightarrow (\neg x_1 \wedge x_2)$
6	$((x_1 \wedge x_3) \vee x_2) \rightarrow ((x_1 \leftrightarrow x_2) \vee x_3)$	21	$(x_1 \vee x_2) \leftrightarrow ((x_1 \rightarrow x_3) \vee \neg x_3)$
7	$((x_1 \wedge x_2) \rightarrow (x_1 \wedge x_3)) \leftrightarrow (x_2 \wedge \neg x_3)$	22	$(\neg(x_1 \wedge x_2) \rightarrow x_3) \leftrightarrow (\neg x_2 \vee x_3)$
8	$((x_1 \vee \neg x_2) \leftrightarrow x_3) \rightarrow ((x_1 \wedge x_2) \leftrightarrow x_3)$	23	$(x_2 \leftrightarrow x_3) \vee ((\neg x_1 \wedge x_3) \rightarrow x_3)$
9	$((x_1 \rightarrow \neg x_2) \wedge x_3) \leftrightarrow ((x_1 \wedge \neg x_2) \vee \neg x_3)$	24	$((\neg x_1 \vee x_2) \wedge x_3) \leftrightarrow (x_1 \rightarrow \neg x_2)$
10	$((\neg x_1 \vee \neg x_2) \leftrightarrow (x_1 \rightarrow x_3)) \vee (x_2 \wedge \neg x_3)$	25	$((\neg x_1 \leftrightarrow \neg x_2) \wedge (x_1 \rightarrow x_3)) \wedge (x_2 \vee x_3)$
11	$(x_1 \vee x_2 \vee x_3) \rightarrow (x_1 \wedge \neg x_3) \vee (x_2 \leftrightarrow x_3)$	26	$(\neg x_1 \vee x_2) \rightarrow (x_1 \wedge \neg x_3) \wedge (x_2 \leftrightarrow x_3)$
12	$(\neg x_1 \rightarrow (x_3 \rightarrow x_2)) \wedge (\neg x_1 \wedge \neg x_3)$	27	$(x_1 \vee (x_2 \leftrightarrow x_3)) \wedge (x_1 \rightarrow \neg x_2)$
13	$(x_3 \rightarrow (x_1 \leftrightarrow x_2)) \leftrightarrow (\neg x_3 \wedge x_1)$	28	$((x_1 \wedge x_2) \rightarrow \neg x_3) \leftrightarrow (\neg x_2 \vee x_3)$
14	$((x_1 \wedge \neg x_2 \wedge x_3) \vee \neg x_2 \vee \neg x_3) \rightarrow x_1$	29	$((x_1 \rightarrow \neg x_2) \leftrightarrow x_3) \rightarrow (x_1 \wedge x_2)$
15	$(x_1 \wedge (x_1 \leftrightarrow \neg x_3)) \rightarrow (x_1 \leftrightarrow \neg x_2)$	30	$(x_2 \rightarrow (\neg x_1 \leftrightarrow x_3)) \wedge (\neg x_1 \vee x_2)$

Ход выполнения работы

1 Используя язык программирования Python (рисунок 3.1), построить таблицу истинности.

2 По таблице истинности найти СДНФ (рисунок 3.2).

+ Код + Текст

Создать таблицу истинности для заданной функции:

$$f(x_1, x_2, x_3) = (x_2 \to (\overline{x_1} \leftrightarrow x_3)) \land (\overline{x_1} \lor x_2)$$

Рисунок 3.1 – Фрагмент отчета по лабораторной работе

+ Код + Текст

```
        if F == 0:
          a = ED(x, y, z)
          D.append(a)
          L.append(a)
        else:
          L.append("-")

        data.append(L)

#вывод таблицы
print(tabulate(data, headers, tablefmt='orgtbl',))
print()
print("СДНФ: F(x1, x2, x3) = " + "V".join(K))
print("СКНФ: F(x1, x2, x3) = " + "∧".join(D))
```

x1	x2	x3	A1	A2	A3	A4	F	EK	ED
0	0	0	1	0	1	1	1	(¬x1∧¬x2∧¬x3)	-
0	0	1	1	1	1	1	1	(¬x1∧¬x2∧x3)	-
0	1	0	1	0	0	1	0	-	(x1∨¬x2∨x3)
0	1	1	1	1	1	1	1	(¬x1∧x2∧x3)	-
1	0	0	0	1	1	0	0	-	(¬x1∨x2∨x3)
1	0	1	0	0	1	0	0	-	(¬x1∨x2∨¬x3)
1	1	0	0	1	1	1	1	(x1∧x2∧¬x3)	-
1	1	1	0	0	0	1	0	-	(¬x1∨¬x2∨¬x3)

```
СДНФ: F(x1, x2, x3) = (¬x1∧¬x2∧¬x3)∨(¬x1∧¬x2∧x3)∨(¬x1∧x2∧x3)∨(x1∧x2∧¬x3)
СКНФ: F(x1, x2, x3) = (x1∨¬x2∨x3)∧(¬x1∨x2∨x3)∧(¬x1∨x2∨¬x3)∧(¬x1∨¬x2∨¬x3)
```

Рисунок 3.2 – Фрагмент отчета по лабораторной работе

Контрольные вопросы

1 Дайте определение булевой функции.
2 Дайте определение элементарной булевой функции второго порядка.
3 Дайте определение СДНФ для булевой функции.
4 Дайте определение СКНФ для булевой функции.

4 Лабораторная работа № 4. Основные классы булевых функций. Минимизация булевых функций

Цель работы: изучение классов булевых функций и их методов минимизации (метод неопределенных коэффициентов, карты Карно, метод Квайна – Мак-Клоски).

Теоретические сведения

Система булевых функций $F=\{f_1, f_2, ..., f_n\}$ называется *полной*, если всякая булева функция является некоторой суперпозицией функций $f_1, f_2, ..., f_n$.

Примерами функционально полных систем могут служить следующие системы:

$$\{\neg, \bullet, \vee\}, \{1, \bullet, \oplus\}, \{\neg, \bullet\}, \{\neg, \vee\}, \{/\}, \{\downarrow\} \text{ и др.}$$

Для определения функциональной полноты системы используются так называемые классы Поста.

1 Класс функций, сохраняющих нуль $\mathbf{P_0}$. Булева функция называется *сохраняющей нуль*, если

$$f(0, 0, ..., 0) = 0.$$

2 Класс функций, сохраняющих единицу $\mathbf{P_1}$. Булева функция называется *сохраняющей единицу*, если

$$f(1, 1, ..., 1) = 1.$$

3 Класс самодвойственных функций **S**. Функция $f^*(x_1, x_2, ..., x_n)$ называется *двойственной* к функции $f(x_1, x_2, ..., x_n)$, если

$$f^*(x_1, x_2, ..., x_n) = \overline{f}\left(\overline{x_1}, \overline{x_2}, ..., \overline{x_n}\right).$$

Булева функция $f(x_1, x_2, ..., x_n)$ называется *самодвойственной*, если

$$f^*(x_1, x_2, ..., x_n) = f(x_1, x_2, ..., x_n).$$

4 Класс монотонных функций **M**. Булева функция называется *монотонной*, если

$$f(\alpha_1, \alpha_2, ..., \alpha_n) \le f(\beta_1, \beta_2, ..., \beta_n)$$

на всех сравнимых наборах, т. е. таких, что $(\alpha_1, \alpha_2, ..., \alpha_n) < (\beta_1, \beta_2, ..., \beta_n)$.

Причем $f(\alpha_1, \alpha_2, ..., \alpha_n) \le f(\beta_1, \beta_2, ..., \beta_n)$, если при любом *i* имеем $\alpha_i \le \beta_i$.

5 Класс линейных функций **L**. Булева функция называется *линейной*, если она представима линейным полиномом Жегалкина.

Метод неопределенных коэффициентов.

Любую булеву функцию от трех переменных можно представить в следующем виде:

$$f(x_1,x_2,x_3) = K_1^1 x_1 \vee K_2^1 x_2 \vee K_3^1 x_3 \vee K_1^0 \bar{x}_1 \vee K_2^0 \bar{x}_2 \vee K_3^0 \bar{x}_3 \vee K_{12}^{11} x_1 x_2 \vee K_{12}^{10} x_1 \bar{x}_2 \vee$$

$$\vee K_{12}^{01} \bar{x}_1 x_2 \vee K_{12}^{00} \bar{x}_1 \bar{x}_2 \vee K_{13}^{11} x_1 x_3 \vee K_{13}^{10} x_1 \bar{x}_3 \vee K_{13}^{01} \bar{x}_1 x_3 \vee K_{13}^{00} \bar{x}_1 \bar{x}_3 \vee K_{23}^{11} x_2 x_3 \vee$$

$$\vee K_{23}^{10} x_2 \bar{x}_3 \vee K_{23}^{01} \bar{x}_2 x_3 \vee K_{23}^{00} \bar{x}_1 \bar{x}_3 \vee K_{123}^{000} \bar{x}_1 \bar{x}_2 \bar{x}_3 \vee K_{123}^{001} \bar{x}_1 \bar{x}_2 x_3 \vee K_{123}^{010} \bar{x}_1 x_2 \bar{x}_3 \vee$$

$$\vee K_{123}^{011} \bar{x}_1 x_2 x_3 \vee K_{123}^{100} x_1 \bar{x}_2 \bar{x}_3 \vee K_{123}^{101} x_1 \bar{x}_2 x_3 \vee K_{123}^{110} x_1 x_2 \bar{x}_3 \vee K_{123}^{111} x_1 x_2 x_3,$$

где коэффициенты K принимают значения «0» или «1».

Чтобы определить коэффициенты *K*, решают следующую систему:

$$\begin{cases}
K_1^0 \vee K_2^0 \vee K_3^0 \vee K_{12}^{00} \vee K_{13}^{00} \vee K_{23}^{00} \vee K_{123}^{000} = f(0,0,0); \\
K_1^0 \vee K_2^0 \vee K_3^1 \vee K_{12}^{00} \vee K_{13}^{01} \vee K_{23}^{01} \vee K_{123}^{001} = f(0,0,1); \\
K_1^0 \vee K_2^1 \vee K_3^0 \vee K_{12}^{01} \vee K_{13}^{00} \vee K_{23}^{10} \vee K_{123}^{010} = f(0,1,0); \\
K_1^0 \vee K_2^1 \vee K_3^1 \vee K_{12}^{01} \vee K_{13}^{01} \vee K_{23}^{11} \vee K_{123}^{011} = f(0,1,1); \\
K_1^1 \vee K_2^0 \vee K_3^0 \vee K_{12}^{10} \vee K_{13}^{10} \vee K_{23}^{00} \vee K_{123}^{100} = f(1,0,0); \\
K_1^1 \vee K_2^0 \vee K_3^1 \vee K_{12}^{10} \vee K_{13}^{11} \vee K_{23}^{01} \vee K_{123}^{101} = f(1,0,1); \\
K_1^1 \vee K_2^1 \vee K_3^0 \vee K_{12}^{11} \vee K_{13}^{10} \vee K_{23}^{10} \vee K_{123}^{110} = f(1,1,0); \\
K_1^1 \vee K_2^1 \vee K_3^1 \vee K_{12}^{11} \vee K_{13}^{11} \vee K_{23}^{11} \vee K_{123}^{111} = f(1,1,1).
\end{cases}$$

Систему начинаем решать с тех уравнений, в которых $f(x_1, x_2, x_3) = 0$. Все коэффициенты, входящие в эти уравнения, также равны нулю. Найденные нулевые коэффициенты подставляем в оставшиеся уравнения системы. Для получения урезанной системы ищем минимальное решение, которое, вообще говоря, не единственное. Минимальное решение системы при подстановке его в функцию дает МДНФ $f(x_1, x_2, x_3)$.

Метод Квайна – Мак-Клоски.

Пусть $f(x_1, ..., x_n) \ne 1$ представлена в СДНФ.

1 *Нахождение первичных импликант.* Все элементарные конъюнкции (ЭК) данной СДНФ сравниваются друг с другом попарно. Если ЭК имеют вид $K \wedge x_i$ и $K \wedge \overline{x_i}$, то выписывается ЭК ранга $(n-1)$. Такие ЭК, для которых произошло «склеивание», помечаются «*». После построения всех ЭК ранга $(n-1)$ вновь

сравнивают их попарно, выписывая ЭК ранга $(n-2)$, полученные после «склеивания», и помечаются «склеенные» ЭК ранга $(n-1)$ и т. д. Этот процесс производится до тех пор, пока некоторые ЭК ранга $l \le n$ уже не будут «склеиваться» между собой.

Все не помеченные ЭК называются *первичными импликантами.*

2 *Расстановка меток. Построение таблицы Квайна.* После выполнения первого этапа получаем

$$f\left(x_1, ..., x_n\right) = \underset{i}{\vee} \lambda_i ,$$

где $f\left(x_1, ..., x_n\right)$ – данная булева функция;

λ_i – ее первичные импликанты.

Строим таблицу, число строк которой равно числу полученных выше первичных импликант $f\left(x_1, ..., x_n\right)$. Число столбцов совпадает с числом ЭК исходной СДНФ $f\left(x_1, ..., x_n\right)$. Если в некоторую ЭК входит первичная импликанта, то на пересечении соответствующих столбца и строки ставится метка. Вносим в таблицу все метки.

3 *Нахождение существенных импликант.* Если в каком-то столбце составленной таблицы имеется только одна метка, то первичная импликанта, стоящая в соответствующей строке, называется *существенной*. Существенная импликанта не может быть исключена из правой части формулы $f\left(x_1, ..., x_n\right)$. Поэтому из таблицы Квайна исключаем строки, соответствующие существенным импликантам, и исключаем столбцы тех ЭК, которые «покрываются» этими существенными импликантами. Иначе говоря, исключаем те столбцы, которые имеют метки на пересечении со строкой существенной импликанты.

4 *Вычеркивание лишних столбцов.* Рассмотрим таблицу, полученную после третьего этапа. Если в этой таблице есть два столбца, в которых метки стоят в одинаковых строках, то один из них вычеркиваем.

5 *Вычеркивание лишних первичных импликант.* В таблице, полученной после четвертого этапа, рассматриваем строки. Если есть строки, в которых отсутствуют метки, то эти строки исключаем из таблицы, а соответствующие первичные импликанты не принимаем далее во внимание.

6 *Выбор минимального покрытия первичными импликантами.* Рассмотрим таблицу, полученную после пятого этапа. Из оставшихся в этой таблице первичных импликант выбираем такую их совокупность, которая содержит в своих строках метки для всех оставшихся столбцов этой таблицы. Из всех возможных вариантов таких совокупностей выбираем тот, для которого общее число переменных в выбранных первичных импликантах наименьшее.

Идея Мак-Клоски. Вместо ЭК ранга $n, n-1, n-2, ...$ используются их двоичные коды. Например, ЭК $x_1 x_2 \overline{x_3}$ имеет код 110. Все коды разбиваются на непересекающиеся группы по количеству единиц в этих кодах. Теперь попарное сравнение можно производить только между соседними по номеру группами.

Вместо исключенных переменных в двоичных кодах ставим прочерк. После выполнения этих действий получаем первичные импликанты.

Метод карт Карно (графическая минимизация булевой функции).

Карта Карно есть не что иное, как форма таблицы для определения булевой функции.

Начнем процесс «склейки»: любые две соседние клетки, содержащие «1», обводятся, и «поглотивший» их прямоугольник представляется словом, содержащим знаки «0», «1» и «×». Причем «×» занимает место той переменной, по которой произведена «склейка». Прямоугольники площадью 2 единицы можно «склеивать» в прямоугольники площадью 4 единицы. Причем таблицу мысленно можно «закручивать» в «цилиндр» по обоим направлениям (т. е. «тор») для нахождения соседних элементов.

Постановка задачи

Выполнить следующие упражнения.

Для булевой функции, заданной в таблице 3.2, найти:

1) минимальную ДНФ методом неопределенных коэффициентов;
2) минимальную ДНФ методом Квайна – Мак-Клоски;
3) минимальную ДНФ методом карт Карно.

Ход выполнения работы

1 Используя язык программирования Python, построить таблицу истинности.

2 Найти минимальную ДНФ заданной булевой функции (рисунки 4.1 и 4.2).

+ Код + Текст

```
    L = []
    for elem in MinFun:
        if MF[i] in elem:  L.append(elem)
    for elem in L:
        MinFun.remove(elem)
    if  len(MinFun)>0:
        MF.append(MinFun[0])
        i +=1

print()
print("Минимальная ДНФ f(x1, x2, x3) = " + " ∨ ".join(MF))
```

Рисунок 4.1 – Фрагмент отчета по лабораторной работе

x1	x2	x3	F
0	0	0	1
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	0

Коэффициенты К:

x1	¬x1	x2	¬x2	x3	¬x3	¬x1¬x2	¬x1x2	x1¬x2
0	1	0	1	0	1	1	0	0
0	1	0	1	1	0	1	0	0
0	1	1	0	0	1	0	1	0
0	1	1	0	1	0	0	1	0
1	0	0	1	0	1	0	0	1
1	0	0	1	1	0	0	0	1
1	0	1	0	0	1	0	0	0
1	0	1	0	1	0	0	0	0

Минимальная ДНФ f(x1, x2, x3) = ¬x1¬x2 ∨ ¬x1x3 ∨ ¬x1x2x3 ∨ x1x2¬x3

Рисунок 4.2 – Фрагмент отчета по лабораторной работе

Контрольные вопросы

1 Дайте определение минимальной ДНФ булевой функции.
2 Дайте определение тупиковой ДНФ булевой функции.
3 Перечислите основные этапы метода неопределённых коэффициентов.
4 Перечислите основные этапы метода Квайна – Мак-Клоски.
5 Приведите пример карты Карно для булевой функции.

5 Лабораторная работа № 5. Основные понятия и определения теории графов

Цель работы: изучение основных понятий теории графов, различных разновидностей графов; построение различных видов.

Теоретические сведения

Граф в общем виде можно определить как совокупность множества V (вершин) и отображения α множества $V \times V$ в некоторое множество M .

Если $M = \mathbb{N}$, граф называется *мультиграфом*; если $M = \{0,1\}$ – *обыкновенным графом*; $M = \mathbb{R}^+$ – *сетью*.

Пары $(a,b) \in V \times V$, для которых $\alpha(a,b) > 0$, называются *рёбрами*. Если $\alpha(a,b) > 1$, то ребро (a,b) называется *кратным*, а граф G, содержащий кратные рёбра, – *мультиграфом*, число $\alpha(a,b)$ – *кратностью ребра*. Если $\forall (a,b)$ $\alpha(a,b) \le 1$, говорят, что граф без кратных рёбер.

Если $|V| = n$ – конечное число, то граф называется *конечным*, а число n – его *порядком*.

Если $\forall (a,b) \in V \times V \quad \alpha(a,b) = \alpha(b,a)$, то граф G называется *неориентированным*. В противном случае граф ориентирован – *орграф*.

Если $\forall a \in V \quad \alpha(a,a) = 0$, то говорят, что граф G не имеет *петель*. В противном случае граф G с петлями.

Обыкновенным называется конечный неориентированный граф без петель и кратных рёбер.

$X \subseteq V \times V$ задаёт множество рёбер графа $G(V,X)$.

Вершины $a,b \in V$ графа $G = (V,X)$ называются *смежными*, если $(a,b) \in X$.

Вершина $a \in V$ и ребро $(b,c) \in X$ называются *инцидентными*, если $a = b$ или $a = c$.

Число $|X|$ называется *размерностью* графа $G = (V,X)$.

Матрица $A = (a_{ij})$, где $a_{ij} = \alpha(a_i, a_j)$ $(a_i, a_j \in V)$, называется *матрицей смежности* графа $G = (V,X)$.

Для неориентированного графа матрица A обладает условием $a_{ij} = a_{ji}$; для графа без петель $a_{ii} = 0$ $(1 \le i \le |V|)$.

Степенью вершины $a \in V$ называется число $\deg a = \left|\{b \in V : (a,b) \in X\}\right|$.

Если $\forall a,b \in V \quad \deg a = \deg b = d$, то граф называется *однородным* (*регулярным*) *степени* d.

Если в графе G имеется k_i вершин степени i, то выражение $\left(1^{k_1} 2^{k_2} \ldots n^{k_n}\right)$ называется *распределением вершин* графа G по степеням.

Полным графом K_n на n вершинах называется граф $G = (V,X)$, у которого $|V| = n$, $X = (V \times V) \setminus \{(a,a), a \in V\}$ ($|X| = C_n^2$).

Кликой графа называется любой его максимальный полный подграф.

Граф $G = (V,X)$ называется *связным*, если $\forall a,b \in V \quad \exists c_1, \ldots, c_k \in V$ $(a,c_1), (c_1,c_2), \ldots, (c_k,b) \in X$.

Максимальный связный подграф графа G называется его *компонентой связности*.

Граф называется *планарным*, если его вершины и рёбра можно уложить в плоскости так, что рёбра не пересекутся.

Хроматическим числом $\operatorname{chr} G$ $(\chi(G))$ графа G называется такое наименьшее положительное число n, что существует отображение множества V на множество $\{1, 2, ..., n\}$ («цветов»), при котором смежные вершины получают разные «цвета».

Граф $G_1 = (V_1, X_1)$ называется *изоморфным* графу $G_2 = (V_2, X_2)$, если существует такое взаимно однозначное отображение $\beta : V_1 \xrightarrow{\beta} V_2$, при котором $\forall a, b \in V_1 \ (a, b) \in X_1 \Longleftrightarrow (\beta a, \beta b) \in X_2 \ (G_1 \cong G_2)$.

В матричном виде $T^T AT = B$, где $A, B -$ матрицы смежности вершин графов G_1 и G_2 соответственно; $T -$ матрица подстановки, соответствующей отображению β.

Автоморфизм графа $G -$ это изоморфизм графа на себя. В матричном виде $T^T AT = A$.

Граф, не содержащий циклов, называется *лесом*.

Дерево − это связный граф без циклов.

Графом n-перестановок назовём граф, вершины которого − все перестановки элементов n-множества и две вершины смежны в том и только том случае, когда одна из них преобразуется в другую транспозицией двух элементов.

Постановка задачи

Выполнить следующие упражнения:

1) сформировать неориентированный граф из 7 вершин;
2) сформировать ориентированный граф из 7 вершин;
3) сформировать взвешенный граф из 7 вершин.

Ход выполнения работы

Используя язык программирования Python, построить графы согласно условию (рисунок 5.1).

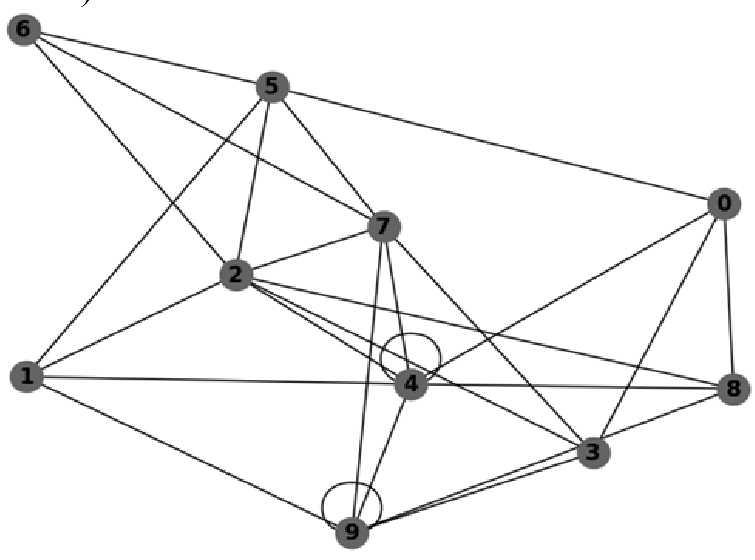


Рисунок 5.1 – Фрагмент отчета по лабораторной работе

Контрольные вопросы

1 Дайте определение графа.

2 Дайте определение ориентированного графа, приведите примеры.

3 Дайте определение неориентированного графа, приведите примеры.

4 Дайте определение взвешенного графа, ориентированного графа, приведите примеры.

6 Лабораторная работа № 6. Способы задания графов

Цель работы: изучение способов задания графов; построение матриц смежности вершин, смежности дуг, инциденций графов.

Постановка задачи

Выполнить следующие упражнения.

Дан орграф $G=(V,U)$, где $V=\{v_1,v_2,v_3,v_4,v_5,v_6\}$ – множество вершин; U – множество дуг. Следует: построить граф; составить матрицу смежности вершин орграфа, матрицу смежности дуг, матрицу инциденций:

1) $U=\{(v_1,v_2),(v_1,v_3),(v_1,v_5),(v_2,v_4),(v_2,v_5),(v_3,v_5),(v_3,v_6),(v_4,v_3),(v_4,v_6),(v_6,v_5)\}$;

2) $U=\{(v_1,v_2),(v_1,v_3),(v_1,v_6),(v_2,v_3),(v_2,v_5),(v_3,v_4),(v_3,v_6),(v_5,v_3),(v_5,v_4),(v_6,v_4)\}$;

3) $U=\{(v_1,v_6),(v_1,v_4),(v_1,v_5),(v_2,v_6),(v_2,v_5),(v_2,v_1),(v_2,v_3),(v_3,v_1),(v_5,v_4),(v_5,v_6)\}$;

4) $U=\{(v_2,v_4),(v_3,v_1),(v_4,v_3),(v_5,v_1),(v_5,v_2),(v_5,v_4),(v_5,v_6),(v_6,v_1),(v_6,v_2),(v_6,v_3)\}$;

5) $U=\{(v_1,v_3),(v_1,v_4),(v_1,v_5),(v_2,v_1),(v_2,v_5),(v_2,v_6),(v_4,v_3),(v_5,v_4),(v_6,v_1),(v_6,v_5)\}$;

6) $U=\{(v_1,v_4),(v_2,v_4),(v_2,v_5),(v_3,v_1),(v_3,v_2),(v_3,v_6),(v_5,v_4),(v_6,v_1),(v_6,v_5),(v_6,v_2)\}$;

7) $U=\{(v_1,v_2),(v_1,v_4),(v_2,v_4),(v_3,v_2),(v_3,v_4),(v_3,v_6),(v_4,v_6),(v_5,v_1),(v_5,v_2),(v_5,v_3)\}$;

8) $U=\{(v_1,v_2),(v_1,v_3),(v_1,v_4),(v_1,v_6),(v_2,v_5),(v_3,v_2),(v_3,v_4),(v_4,v_2),(v_6,v_3),(v_6,v_2)\}$;

9) $U=\{(v_1,v_6),(v_2,v_3),(v_2,v_5),(v_2,v_4),(v_3,v_1),(v_4,v_1),(v_4,v_3),(v_5,v_1),(v_5,v_3),(v_5,v_4)\}$;

10) $U=\{(v_2,v_4),(v_2,v_6),(v_2,v_5),(v_3,v_2),(v_4,v_1),(v_5,v_1),(v_5,v_6),(v_5,v_4),(v_6,v_1),(v_6,v_4)\}$.

Ход выполнения работы

Используя язык программирования Python, построить графы согласно условию (рисунок 6.1).

```

# добавляем информацию в объект графа
G.add_nodes_from(nodes)
G.add_edges_from(edges)

# рисуем граф и отображаем его
nx.draw(G, with_labels=True, font_weight='bold')
plt.show()
```

```
Список вершин:
[0, 1, 2, 3, 4, 5, 6, 7, 8, 9]

Матрица смежности вершин:
0,0,0,1,1,1,0,0,1,0
0,0,1,0,1,1,0,0,0,1
0,1,0,1,1,1,1,1,1,0
1,0,1,0,0,0,0,1,0,1
1,1,1,0,1,0,0,1,1,1
1,1,1,0,0,0,1,1,0,0
0,0,1,0,0,1,0,1,0,0
0,0,1,1,1,1,1,0,0,1
1,0,1,0,1,0,0,0,0,1
0,1,0,1,1,0,0,1,1,1

Список ребер:
0 :(0, 3)(0, 4)(0, 5)(0, 8)
1 :(1, 2)(1, 4)(1, 5)(1, 9)
2 :(2, 1)(2, 3)(2, 4)(2, 5)(2, 6)(2, 7)(2, 8)
3 :(3, 0)(3, 2)(3, 7)(3, 9)
4 :(4, 0)(4, 1)(4, 2)(4, 4)(4, 7)(4, 8)(4, 9)
5 :(5, 0)(5, 1)(5, 2)(5, 6)(5, 7)
6 :(6, 2)(6, 5)(6, 7)
7 :(7, 2)(7, 3)(7, 4)(7, 5)(7, 6)(7, 9)
8 :(8, 0)(8, 2)(8, 4)(8, 9)
9 :(9, 1)(9, 3)(9, 4)(9, 7)(9, 8)(9, 9)
```

Рисунок 6.1 – Фрагмент отчета по лабораторной работе

Контрольные вопросы

1 Дайте определение матрицы смежности вершин графа.
2 Дайте определение матрицы смежности дуг графа.
3 Дайте определение матрицы ициденций графа.

7 Лабораторная работа № 7. Операции над графами. Метрические характеристики графов

Цель работы: изучение операций над графами, метрических характеристик графов и способов их определения.

Теоретические сведения

Хроматическим числом $\mathrm{chr}\,G$ $(\chi(G))$ графа G называется такое наименьшее положительное число n, что существует отображение множества V на множество $\{1, 2, ..., n\}$ («цветов»), при котором смежные вершины получают разные «цвета».

Граф $G_1 = (V_1, X_1)$ называется *изоморфным* графу $G_2 = (V_2, X_2)$, если существует такое взаимно однозначное отображение $\beta : V_1 \xrightarrow{\beta} V_2$, при котором $\forall a, b \in V_1$ $(a, b) \in X_1 \Longleftrightarrow (\beta a, \beta b) \in X_2$ $(G_1 \cong G_2)$.

В матричном виде $T^T AT = B$, где $A, B -$ матрицы смежности вершин графов G_1 и G_2 соответственно; $T -$ матрица подстановки, соответствующей отображению β.

Автоморфизм графа $G -$ это изоморфизм графа на себя. В матричном виде $T^T AT = A$.

Постановка задачи

Для графа, полученного в лабораторной работе № 6, выполнить следующие упражнения:

1) добавить вершину графа, соединив ее с тремя произвольными вершинами;
2) удалить произвольную вершину графа;
3) определить метрические характеристики полученного графа.

Составить матрицу смежности вершин орграфа, матрицу смежности дуг, матрицу инциденций и каждом пункте.

Ход выполнения работы

Используя язык программирования Python, построить графы согласно условию (рисунки 7.1 и 7.2).

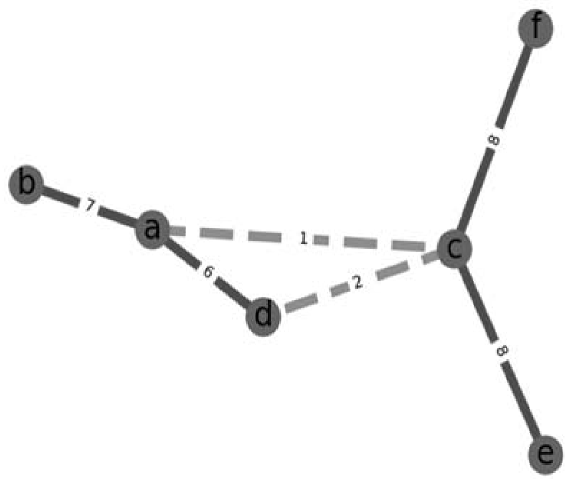


Рисунок 7.1 – Фрагмент отчета по лабораторной работе

+ Код + Текст

```
nx.draw_networkx_nodes(G, pos, node_size = Diameter)

# edges
nx.draw_networkx_edges(G, pos, edgelist = elarge, width = 6, edge_color = "r")
nx.draw_networkx_edges(G, pos, edgelist = esmall, width=6, alpha = 0.5, edge_color = "b", style="dashed")

# node labels
nx.draw_networkx_labels(G, pos, font_size=20, font_family = "sans-serif")
# edge weight labels
edge_labels = nx.get_edge_attributes(G, "weight")
nx.draw_networkx_edge_labels(G, pos, edge_labels)

ax = plt.gca()
ax.margins(0.08)
plt.axis("off")
plt.tight_layout()
plt.show()
```

Рисунок 7.2 – Фрагмент отчета по лабораторной работе

Контрольные вопросы

1 Дайте определение хроматического числа графа.
2 Дайте определение изоморфного графа.
3 Приведите пример операции: добавление вершины графа.
4 Приведите пример операции: удаление вершины графа.

8 Лабораторная работа № 8. Упорядочение элементов графа

Цель работы: изучение различных упорядочений графов, алгоритма построения сети их метрических характеристик.

Теоретические сведения

Упорядоченный граф – граф, изоморфный исходному, для которого характерны следующие правила:

1) вершины графа расположены по слоям;
2) вершины первого слоя не имеют предшествующих вершин;
3) вершины последнего слоя не имеют последующих вершин;
4) вершины i-го слоя соединяются дугами с вершинами (i + 1)-го слоя;
5) вершины, находящиеся на одном слое, не соединяются.

Упорядочим граф, представленный на рисунке 8.1.

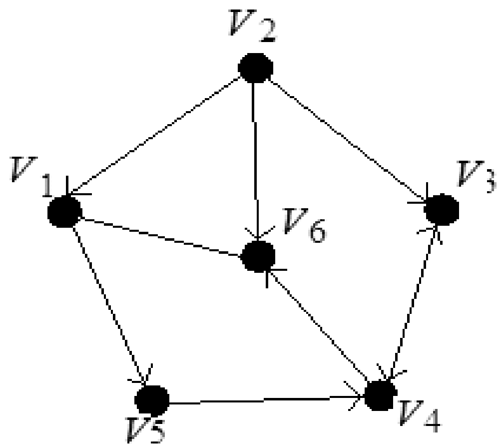


Рисунок 8.1 – Граф

Рассмотрим алгоритм упорядочения графа.

1 Находим в графе вершины, в которые не входит ни одна дуга. Они образуют первый слой. Удаляем из графа эти вершины и дуги, из них исходящие.

2 В полученном графе находим вершины, в которые не входит ни одна дуга. Получаем вершины второго слоя. Удаляем эти вершины и дуги, из них исходящие.

3 Аналогично для остальных вершин повторяем п. 2 алгоритма.

4 Процесс продолжается до тех пор, пока из исходного графа не будут удалены все вершины и дуги (рисунок 8.2).

Аналогичным способом можно упорядочить не только вершины, но и дуги графа.

Упорядочение производится по матрице смежности вершин графа.

1 Строим вектор x_1, компонентами которого является сумма элементов столбцов матрицы смежности.

2 В этом векторе x_1 находим нулевую координату, т. е. вершину, соответствующую нулевой компоненте, находящейся на первом слое.

3 Из матрицы смежности вычёркиваем строку, соответствующую вершинам первого слоя.

4 Находим компоненты вектора x_2 (сумму столбцов без вычеркнутой строки).

5 Нулевые компоненты вектора x_2 образуют вершины второго слоя.

6 Вычёркивая из матрицы строки, соответствующие вершинам второго слоя, находим компоненты вектора x_3 и т. д.

7 Процесс продолжается до тех пор, пока все строки матрицы не будут вычеркнуты.

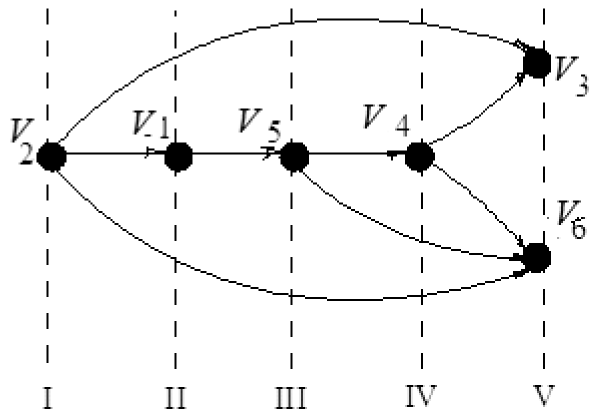


Рисунок 8.2 – Упорядоченный граф

Постановка задачи

Задание 1

Упорядочить вершины орграфа и построить изоморфный граф, заданный в условии лабораторной работы № 6.

Теоретические сведения

Сетью называют ориентированный связный граф $G(V,E)$ с множеством вершин V и множеством дуг E, где каждой дуге $(v_i,v_j)\in E$ придается числовая характеристика r_{ij}, называемая *пропускной способностью*. В G выделяют две фиксированные вершины – s и t; s называют *истоком*, t – *стоком*, а остальные вершины – *промежуточными*. Пропускная способность дуги $r_{ij}\ge 0$, и если $(v_i,v_j)\notin E$, то $r_{ij}=0$.

Множество чисел $X=\{x_{ij}\}$ называют *потоком по сети*, а сами x_{ij}, определенные на дугах $(v_i,v_j)\in E$, – *потоками в дугах*, если выполняются следующие условия: $x_{ij}=-x_{ji}$; $x_{ij}\le r_{ij}$ (поток в дуге (v_i,v_j) не может превышать пропускной способности этой дуги); $\sum_{j=1}^{n} x_{ij} = 0\left(v_i \neq s,t\right)$ (условие сохранения потока в промежуточных вершинах).

Если $x_{ij} < r_{ij}$ то дуга (v_i, v_j) называется *ненасыщенной*, если же $x_{ij} = r_{ij}$ – *насыщенной*.

Суммарный поток из истока s равен суммарному потоку в сток t, т. е.

$$f = \sum_{j=1}^{n} x_{kj} = \sum_{i=1}^{n} x_{il} \left(v_k = s, v_l = t\right),$$

где v_j – конечные вершины дуг, исходящих из s;

v_i – начальные вершины дуг, входящих в t.

Функцию f называют *мощностью потока* на сети.

Разрез сети $S/\overline{S}$ – множество дуг, для которых выполняются следующие требования: $s \in S$, $t \in \overline{S}$; $S \cap \overline{S} = \varnothing$; $S \cup \overline{S} = V$.

Пропускная способность разреза $R\left(S/\overline{S}\right)$ равна сумме пропускных способностей r_{ij} всех дуг, входящих в разрез.

Поток через разрез $X\left(S/\overline{S}\right)$ равен сумме потоков x_{ij} по всем дугам, входящих в разрез.

Теорема Форда – Фалкерсона: максимальный поток по заданной сети равен минимальной пропускной способности разреза, отделяющего s от t.

Алгоритм расчета максимального потока.

1 Выписываем полный путь из истока в сток, определяя при этом пропускную способность. Пропускная способность равна наименьшему из значений пропускных способностей дуг, составляющих этот путь. Над каждой дугой пути записываем поток, равный пропускной способности данного пути. Если поток и пропускная способность совпадают, то *дуга насыщенная*, выделяем её.

2 Строим другие полные пути и считаем их пропускную способность до тех пор, пока сток достигаем из истока по ненасыщенным дугам. Если какая-либо дуга пути входит в другой путь, то пропускная способность данной дуги равна первоначальной пропускной способности минус поток по другому пути. К потоку данной дуги прибавляем поток по новому пути.

3 Определяем множество S как исток s и вершины, достигаемые из истока по ненасыщенным дугам. Тогда множество $\overline{S}$ образует остальные вершины сети. Выделяем разрез на сети (т. е. множество дуг сети (v_i, v_j) таких, что $v_i \in S$, а $v_j \in \overline{S}$).

4 Считаем мощность потока сети f и пропускную способность разреза $R\left(S/\overline{S}\right)$. В соответствии с теоремой Форда – Фалкерсона эти величины равны друг другу.

Пример – Рассчитать максимальный поток в сети (рисунок 8.3) от истока $v_1 = s$ до стока $v_9 = t$.

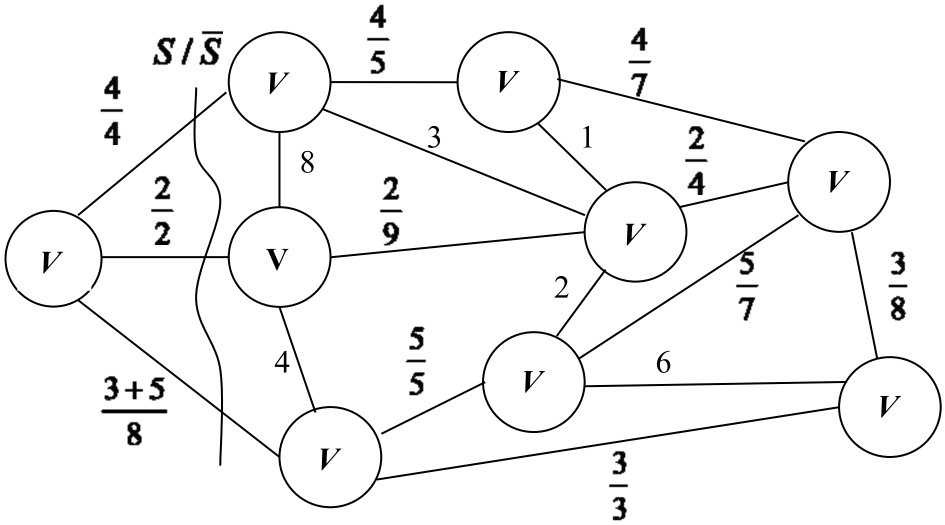


Рисунок 8.3 – Сеть

Решение

Запишем полные пути от истока к стоку, их пропускную способность и потоки:

$L_1 = (v_1, v_2, v_5, v_9)$, $R(L_1) = \min\{r_{12}, r_{25}, r_{59}\} = \min\{4,5,7\} = 4$, $x_{12} = x_{25} = x_{59} = 4$;

$L_2 = (v_1, v_4, v_8, v_9)$, $R(L_2) = \min\{r_{14}, r_{48}, r_{89}\} = \min\{8,3,8\} = 3$, $x_{14} = x_{48} = x_{89} = 3$;

$L_3 = (v_1, v_3, v_7, v_9)$, $R(L_3) = \min\{r_{13}, r_{37}, r_{79}\} = \min\{2,9,4\} = 2$, $x_{13} = x_{37} = x_{79} = 2$;

$L_4 = (v_1, v_4, v_6, v_9)$, дуга (v_1, v_4) входит в путь L_2, поэтому $R(L_4) = \min\{r_{14} - x_{14}, r_{46}, r_{69}\} = \min\{8-3,5,7\} = 5$, $x_{14} = 3+5 = 8, x_{46} = x_{69} = 5$.

Здесь $S = \{v_1\}$, $\bar{S} = \{v_2, v_3, v_4, v_5, v_6, v_7, v_8, v_9\}$. Разрез определяем как $S / \bar{S} = \{(v_1, v_2), (v_1, v_3), (v_1, v_4)\}$. Мощность потока $f = x_{12} + x_{13} + x_{14} =$ $= 4 + 2 + 8 = 14$. Пропускная способность разреза $R\left(S / \bar{S}\right) = r_{12} + r_{13} + r_{14} =$ $= 4 + 2 + 8 = 14.$

Постановка задачи

Рассчитать максимальный поток на заданной сети (рисунок 8.4). Веса ребер сети выбрать произвольным образом от 5 до 10.

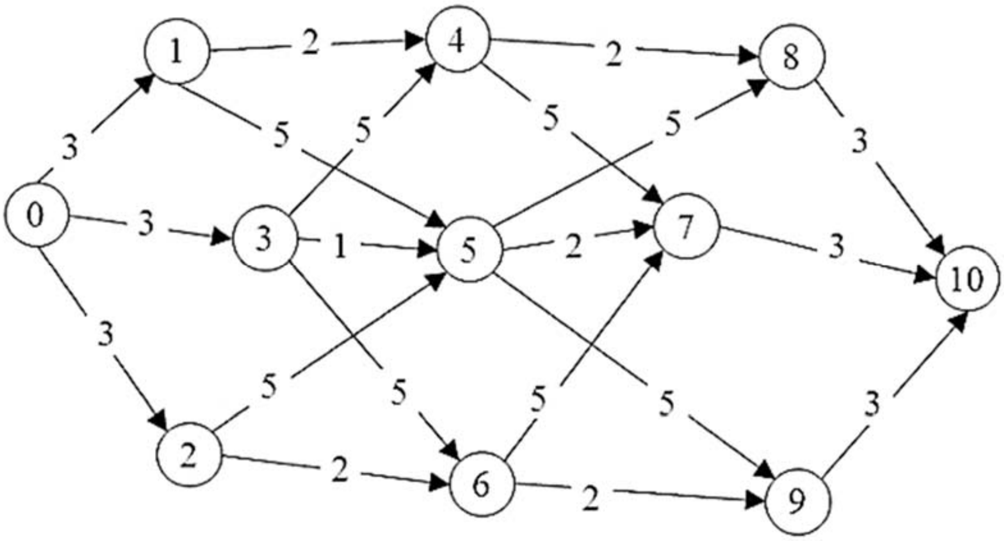


Рисунок 8.4 – Сеть

Ход выполнения работы

Используя язык программирования Python, построить графы согласно условию (рисунок 8.5).

```
edge_labels = nx.get_edge_attributes(G, "weight")
nx.draw_networkx_edge_labels(G, pos, edge_labels)

ax = plt.gca()
#ax.margins(0.08)
plt.axis("off")
plt.tight_layout()
plt.show()
```

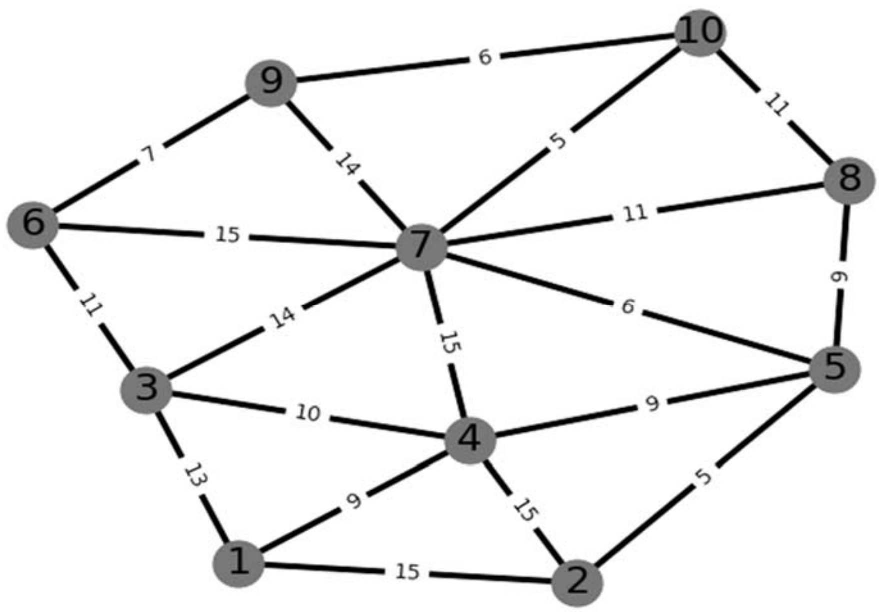


Рисунок 8.5 – Фрагмент отчета по лабораторной работе

Контрольные вопросы

1 Приведете пример графического упорядочения графа.

2 Приведете пример упорядочения графа, используя матричный метод.

3 Приведите пример сети, содержащей 4–5 узлов (вершин).

4 Определите максимальный поток по сети, полученной в предыдущем вопросе.

5 Определите разрез на сети, полученой в п. 3.

Список литературы

1 **Баврин, И. И.** Дискретная математика: учебник и задачник для вузов / И. И. Баврин. – М. : Юрайт, 2024. – 193 с.

2 **Гашков, С. Б.** Дискретная математика : учебник и практикум для среднего проф. образования / С. Б. Гашков, А. Б. Фролов. – 4-е изд., перераб. и доп. – М. : Юрайт, 2024. – 530 с.

3 **Микони, С. В.** Дискретная математика для бакалавра: множества, отношения, функции, графы : учеб. пособие / С. В. Микони. – СПб.; М. ; Краснодар : Лань, 2021. – 192 с.

4 **Поздняков**, **С. Н.** Дискретная математика : учебник для вузов / С. Н. Поздняков. – М. : Академия, 2008. – 448 с.