МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ

РОССИЙСКОЙ ФЕДЕРАЦИИ

федеральное государственное бюджетное образовательное учреждение

высшего образования

«Российский государственный университет им. А.Н. Косыгина

(Технологии. Дизайн. Искусство)»

Кафедра автоматизированных систем обработки информации и управления

Отчет по лабораторной работе № 1

по дисциплине «Теория множеств и алгоритмы на графах»

по теме: «Операции над множествами»

Выполнил: Ольховский Н.С., ИТА-123

Проверила: Кузьмина Т.М.

Москва, 2025

# Вариант 13

# Задание 1

1. Написать и отладить программу, которая позволяет выполнить действия, задаваемые формулой, определенной вариантом задания. Множества можно задать один раз внутри программного кода, универсальное множество определяется как объединение всех заданных множеств.  
   На экран должны быть выведены сами множества и результаты вычислений «по действиям». Для каждого действия определяется отдельная кнопка и отдельное поле вывода.
2. Разработать тестовый пример :
   1. Задать 4 множества натуральных чисел A, B, C, D. В формуле может присутствовать 3 множество, но задать надо все 4 множества. Значения, для задания множеств, нужно брать на интервале [10\*n,10\*n+20] , где n - номер варианта. Задавать множества можно вручную, а можно с помощью генератора случайных чисел. Все множества должны попарно пересекаться, так же должны пересекаться все тройки множеств. Пересечение всех 4-х множеств может быть пустым.
   2. Универсальное множество определяется, как объединение всех заданных множеств.
3. Запустить программу на тестовом примере, сделать скрины, демонстрирующие работу программы по вычислению формулы.
4. В отчете для каждого скрина, на котором показан результат вычислений по определенной формуле, построить диаграмму Эйлера-Венна.
5. Обязательно указать, каким способом закрашено итоговое множество. На диаграмму нанести числа – элементы тестовых множеств.

# 

# Текст программы

*# Множества A, B, C, U (универсальное)*

A = set([130, 133, 135, 137, 139])

B = set([133, 136, 137, 140, 142])

C = set([135, 136, 137, 141, 143])

D = set([137, 138, 139, 140, 144])

U = A.union(B).union(C).union(D)

print("Множества:")

print("A:", A)

print("B:", B)

print("C:", C)

print("D:", D)

print(f"U (универсальное): {U}\n")

*# Дополнения*

input("Для вычисления дополнений нажмите Enter")

notA = U.difference(A)

notB = U.difference(B)

notC = U.difference(C)

print(f"ㄱA: {notA} \nㄱB: {notB} \nㄱC: {notC}\n")

*# Пересечения*

input("Для вычисления пересечений нажмите Enter")

A\_and\_notC = A.intersection(notC)

notB\_and\_C\_and\_notA = notB.intersection(C).intersection(notA)

print(f"A ∩ ㄱC: {A\_and\_notC} \nㄱB ∩ C ∩ ㄱ A: {notB\_and\_C\_and\_notA}\n")

*# Объединение*

input("Для вычисления объединения нажмите Enter")

A\_and\_notC\_or\_notB\_and\_C\_and\_notA = A\_and\_notC.union(notB\_and\_C\_and\_notA)

print("A ∩ ㄱC ∪ ㄱB ∩ C ∩ ㄱ A:", A\_and\_notC\_or\_notB\_and\_C\_and\_notA)

# Работа программы

Скриншот работы представлен на рисунке 1.

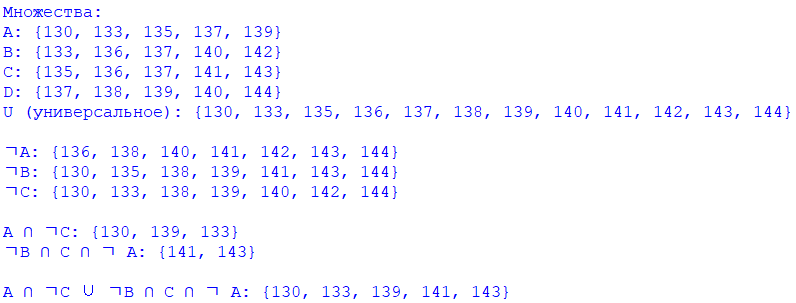


Рис. 1 – Вывод результата программы

# Диаграммы Эйлера-Венна

Диаграммы Эйлера-Венна представлены на рисунке 2.

|  |  |  |
| --- | --- | --- |
| ㄱA | ㄱB | ㄱC |
|  |  |  |
| A ∩ ㄱC | ㄱB ∩ C ∩ ㄱ A | A∩ㄱC∪ㄱB∩C∩ㄱA |
|  |  |  |

Рис. 2 – Диаграммы Эйлера-Венна

# Задание 2

1. Построить диаграмму Эйлера-Венна для множества, заданного формулой, расположенной слева от знака .
2. Построить диаграмму Эйлера-Венна для множества, заданного формулой, расположенной справа от знака .
3. Сравнить итоговые множества обеих формул и сделать выводы.

Формула варианта: 

# Диаграммы Эйлера-Венна

Диаграммы Эйлера-Венна представлены на рис. 3.

|  |  |
| --- | --- |
| (B-C) | A-(B-C) |
|  |  |
| B ∩ C | A ∪ B ∩ C |
|  |  |

Рис. 3 – Диаграммы Эйлера-Венна

**Вывод:** Формула верна, A-(B-C) входит в A ∪ B ∩ C за исключением A∪B∪ㄱC