Министерство науки и высшего образования Российской Федерации

Федеральное государственное автономное образовательное учреждение

высшего образования

**«Пермский национальный** **исследовательский политехнический университет»**

Электротехнический факультет

Кафедра «Информационные технологии и автоматизированные системы»

направление подготовки: 09.03.01 – «Информатика, вычислительная техника»

**О Т Ч Е Т**

**по лабораторной работе №1 на тему:  
 «Калькулятор множеств»**

Выполнил студент гр. ИВТ-23-1б

Пискунов Дмитрий Александрович

(Фамилия, Имя, Отчество)

(подпись)

Проверил:

ст. преп. каф. ИТАС

Рустамханова Гульшат Ильдаровна

(должность, Ф.И.О руководителя от кафедры)

(оценка) (подпись)

(дата)

Пермь 2024

# СОДЕРЖАНИЕ

[СОДЕРЖАНИЕ 2](#_Toc179503931)

[Введение 3](#_Toc179503932)

[Описание работы программы 4](#_Toc179503933)

[Заключение 10](#_Toc179503934)

[Список использованных источников 11](#_Toc179503935)

# Введение

В результате лабораторной работы необходимо реализовать калькулятор множеств, выполняющий следующие функции:

1. Предоставление возможности задать более 3 множеств
2. Множества должны быть в универсуме, включающем в себя целые числа от -50 до 50
3. Множества можно задать следующими способами:

3.1) Случайными числами

3.2) Вручную

3.3) По совокупности условий (соблюдение знака, кратность числу, нахождение в диапазоне)

4) Позволение написать формулу

# 

# Описание работы программы

Для написания программы части был выбран язык Python

Для работы с множествами и их хранения был использован список списков, изначальный размер которого задаётся введенным пользователем числом (рисунок 1)

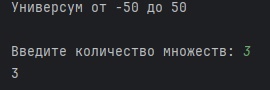


Рисунок 1 – ввод количества множеств

Далее через цикл происходит заплнение каждого множества, вначале запрашивается количество элементов множества, затем предлагается выбрать способ заполнения: вручную либо псевдослучайными числами (рисункок 2).

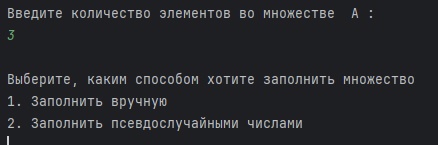


Рисунок 2 – выбор способа зполнения множества

При выборе первого пункта каждый элемент множества вводится пользователем вручную (рисунок 3).

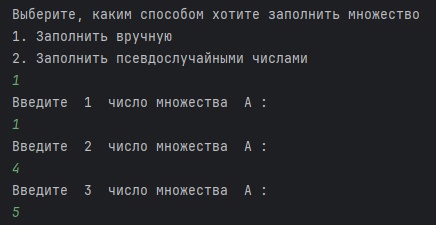


Рисунок 3 – заполнения множества в ручную

При выборе второго пункта предлагается ряд условий по которым будут генерироваться элементы множества (рисунок 4).

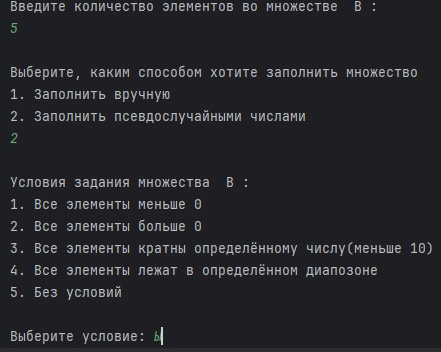


Рисунок 4 – выбор условия при заполнении псевдослучайнми числами

Кратность определённому числу не должна превышать 10 для возможности заполнения больших множеств. Ниже представлены результаты заполнения множеств с каждым условием ( рисунки 5, 6, 7, 8, 9, 10)

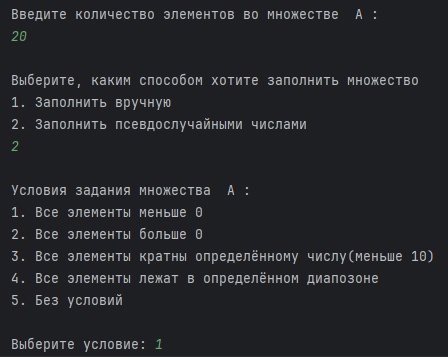


Рисунок 5 – заполнение множества псевдослучайнми числами меньше 0

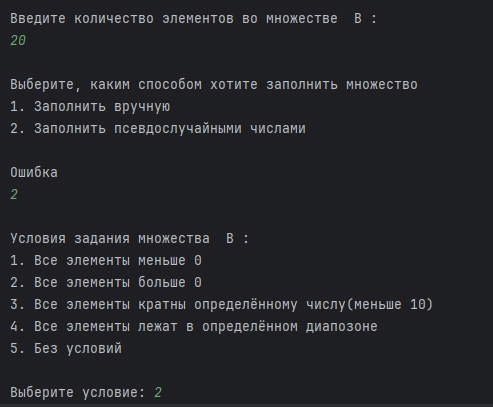


Рисунок 6 – заполнение множества псевдослучайными числами больше 0

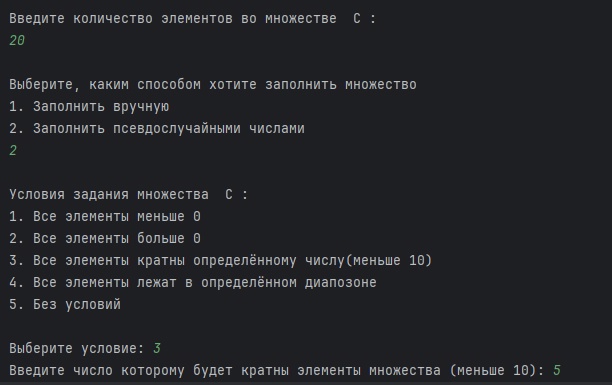


Рисунок 7 – заполнение множества псевдослучайными числами кратными 5

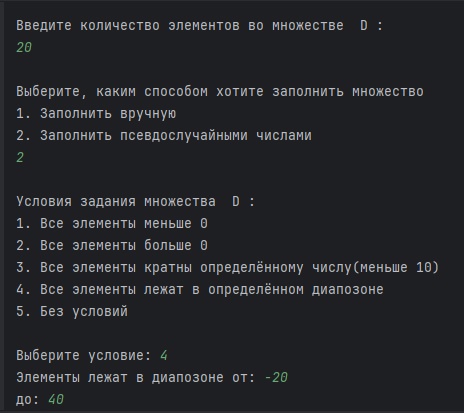


Рисунок 8 – заполнение множества псевдослучайными числами в диапозоне от -20 до 40

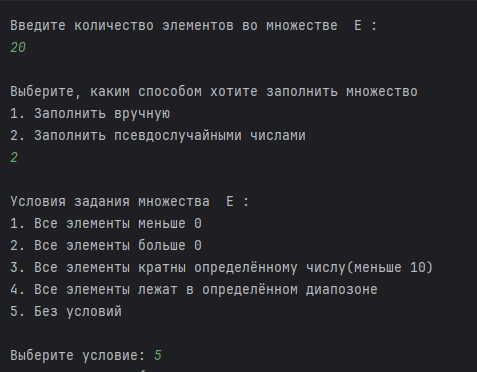


Рисунок 9 – заполнение множество псевдослучайными числами без условий

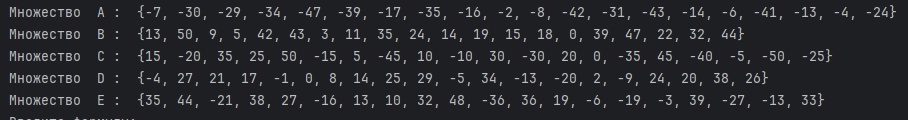


Рисунок 10 – результаты заполнений

Далее предлагается ввести формулу с заданными правилами(рисунок 11)

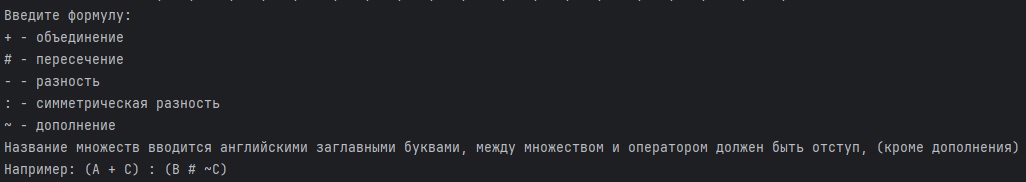


Рисунок 11 – правила ввода формулы

Решение формулы осуществляется через стек, в который помещается каждый символ с конца не считая пробелов, далее достается поочередно достаётся по 2 множества, знак между ними и дополнение если оно есть. Сначала находится дополнение множества, после чего уже действие написанное в формуле, резульатом является еще одно множество, которое также помещается в стек. Решение происходит до того момента, пока в стеке не останется 1 множество, которое и будет результатом решения. Если в формуле есть скобки, то снчала решаются они. На рисунке 12 предстален ввод формулы и результат её решения

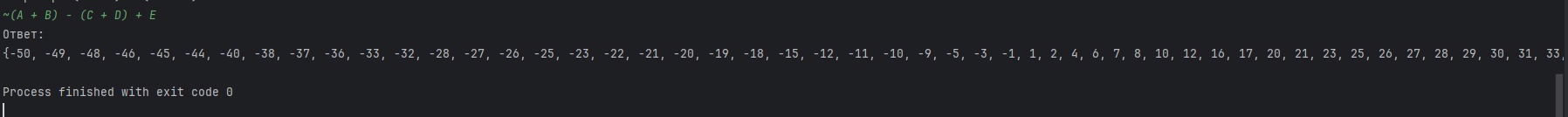


Рисунок 12 – ввод формулы и резульать её решения

Ниже предоставлены результаты выполнения всех действий над множествами (рисунки 13, 14, 15, 16, 17)

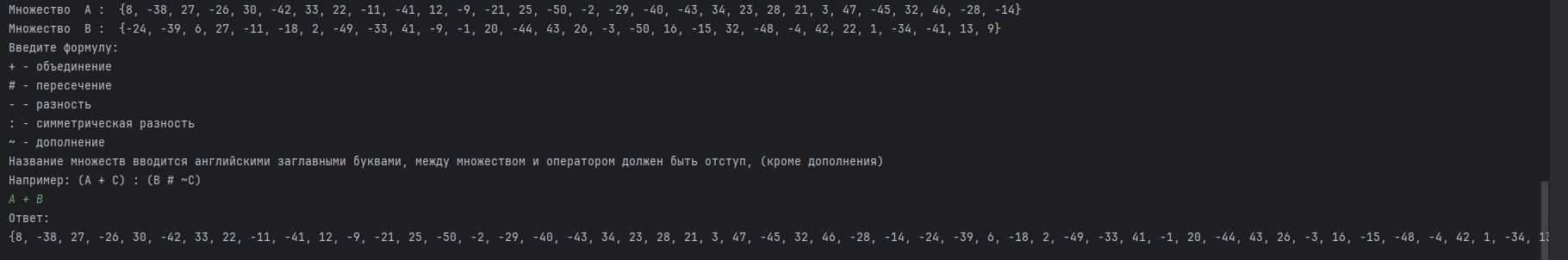


Рисунок 13 – результат объединения двух множеств

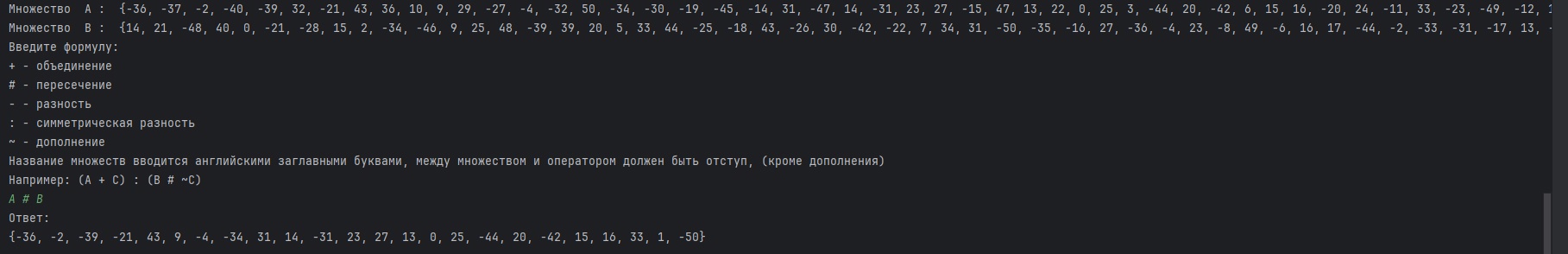


Рисунок 14 – результат пересечения двух множеств

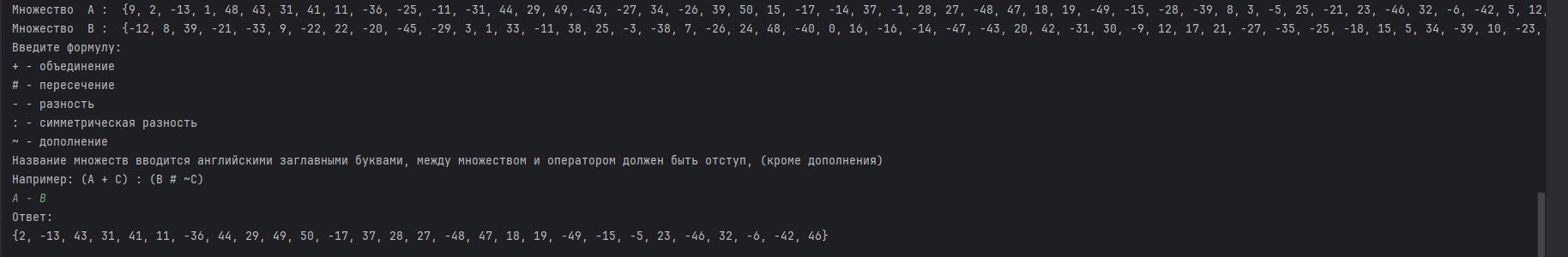


Рисунок 15 – результат разности двух множеств

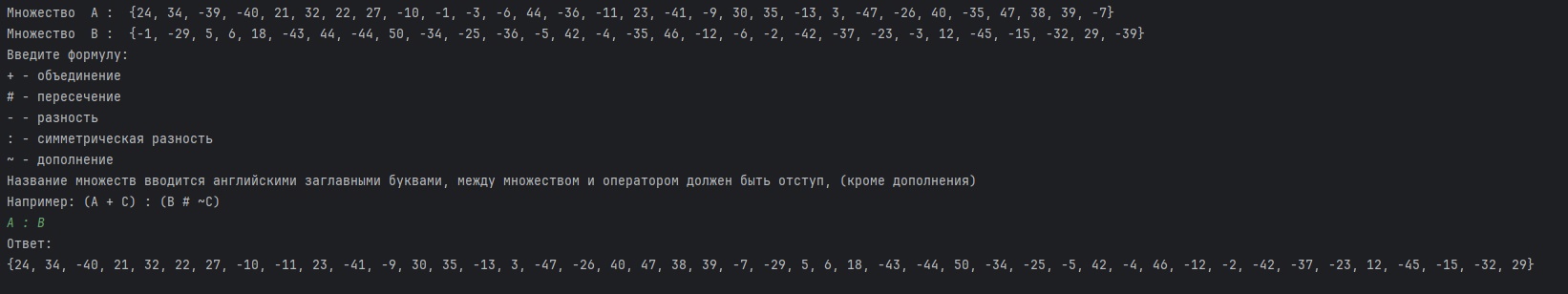


Рисунок 16 – результат симметрической разности двух множеств

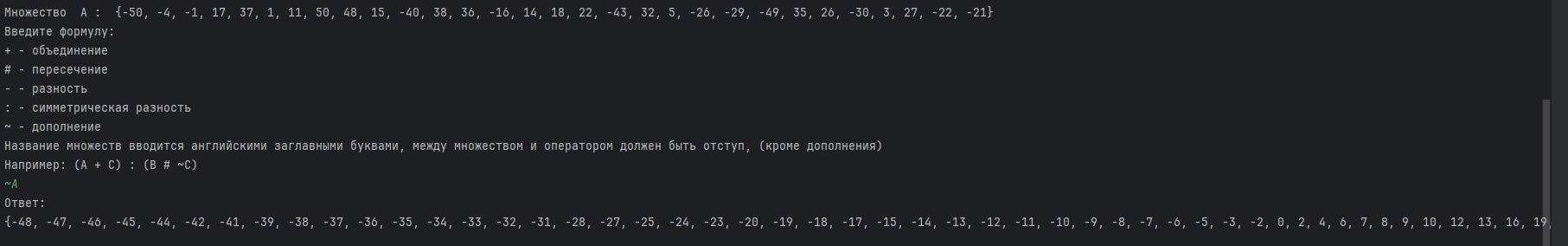


Рисунок 17 – результат дополнения множества

На каждом этапе выполнения программы есть проверка на неверный ввод пользователя. При неверном воде программе выведет ошибку и повторно попросит ввод, пример предоставлен на рисунке 18.

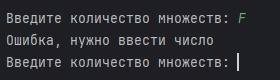


Рисунок 18 – поведение программы при неверном вводе

# Заключение

В ходе лабораторной работы был реализован калькулятор множеств, который выполняет все поставленные задачи: предоставляет возможность задать более 3 множеств, множество входят в универсум представляющий из себя целые числа от -50 до 50, множества можно задать случайными числами, вручную, и по совокупности условий и позволяет написать формулу.

Исходный код готовой программы представлен на GitHub по адресу <https://github.com/MOkASiH/Math/tree/main/1_Lab>. Программа была протестирована и отлажена, после чего не имеет серьёзных ошибок и недочетов.

При выполнении этой лабораторной работы были получены навыки работы с множествами.

# Список использованных источников

1. 8. Errors and Exception – Python 3.13.0 documentation URL: <https://docs.python.org/3/tutorial/errors.html> (дата обращения: 27.09.2024).
2. 5. Data Structures – Python 3.13.0 documentation URL: <https://docs.python.org/3/tutorial/datastructures.html> (дата обращения: 27. 09.2024)
3. Функции в Python – вызов функций def, return, аргументы и параметры URL: <https://pythonchik.ru/osnovy/funkcii-v-python> (дата обращения 29.09.2024)