Министерство науки и высшего образования Российской Федерации

федеральное государственное бюджетное образовательное учреждение высшего образования

«Российский экономический университет имени Г.В. Плеханова»

**МОСКОВСКИЙ ПРИБОРОСТРОИТЕЛЬНЫЙ ТЕХНИКУМ.**

Специальность: 09.02.07 Информационные системы и программирование

Квалификация: Программист

ПРАКТИЧЕСКАЯ РАБОТА №1

ПО «ОСНОВЫ АЛГОРИТМИЗАЦИИ И ПРОГРАММИРОВАНИЯ»

ТЕМА: Калькулятор в PYTHON

Листов: 3

|  |  |
| --- | --- |
| Выполнил студент  Группы: П-5-24  Трусов Дмитрий Олегович | Проверил преподаватель  А.А Никонова  «\_\_\_\_» \_\_\_\_\_\_\_2025 года |

Москва 2025

Цель работы:Разработать калькулятор на Python, который выполняет операции с числами на основе введенных пользователем данных. Калькулятор должен поддерживать арифметические операторы, операторы сравнения, логические и побитовые операторы, а также операторы принадлежности и тождественности. Программа должна включать обработку ошибок для обеспечения корректной работы при вводе неверных данных или выполнении некорректных операций (например, деления на ноль).

Ход работы:

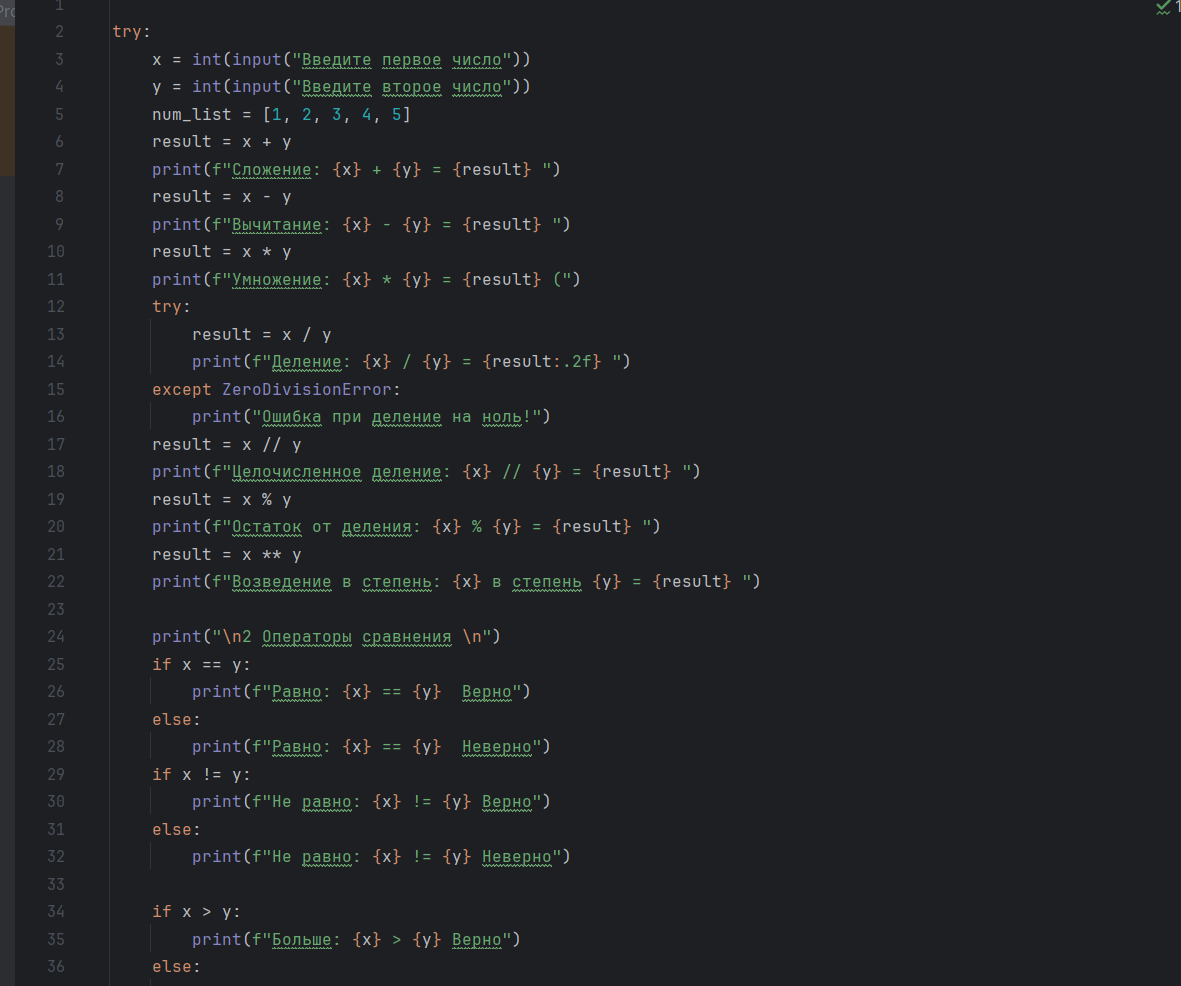


Рисунок 1 Ввод пользователя двух чисел

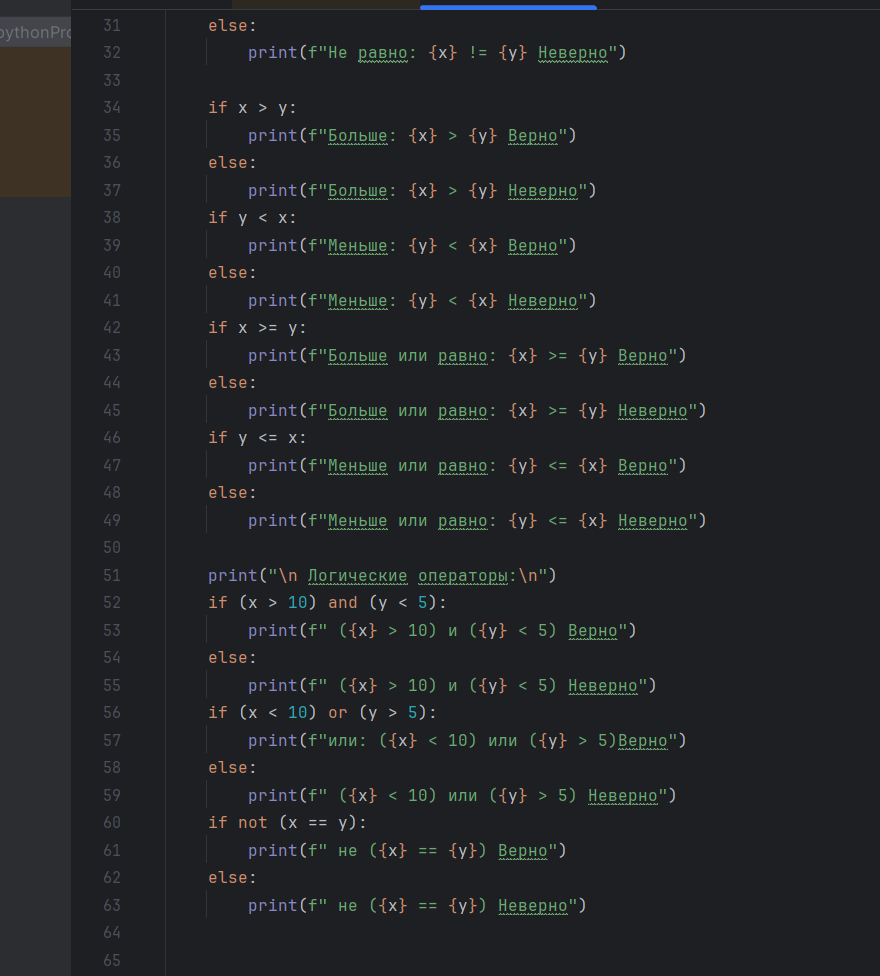
Для реализации калькулятора используется язык программирования Python. Программа начинается с запроса у пользователя двух чисел (а для некоторых операций — одного числа или проверки на принадлежность списку). 

Рисунок 2 Сравнение чисел и логические операторы

Сначала программа обрабатывает арифметические операции: сложение, вычитание, умножение, деление (с проверкой деления на ноль), целочисленное деление, остаток от деления и возведение в степень. Затем выполняются операции сравнения: равно, не равно, больше, меньше, больше или равно, меньше или равно. Для логических операций проверяются условия с использованием AND, OR и NOT.

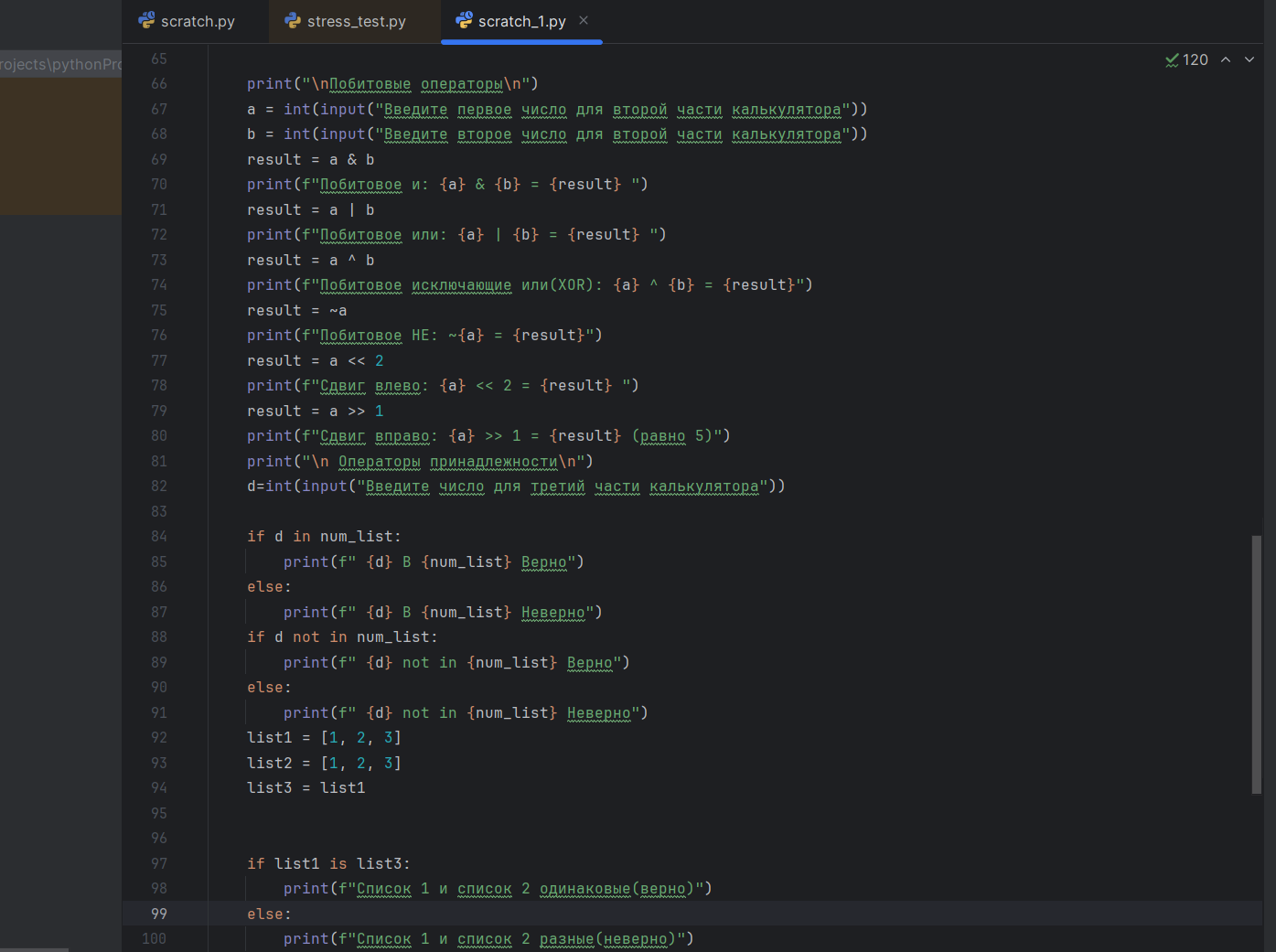


Рисунок 3 Побитовые операции

Далее программа переходит к побитовым операциям: AND, OR, XOR, NOT, сдвиг влево и сдвиг вправо. Для этого пользователь вводит два числа, над которыми производятся побитовые действия.

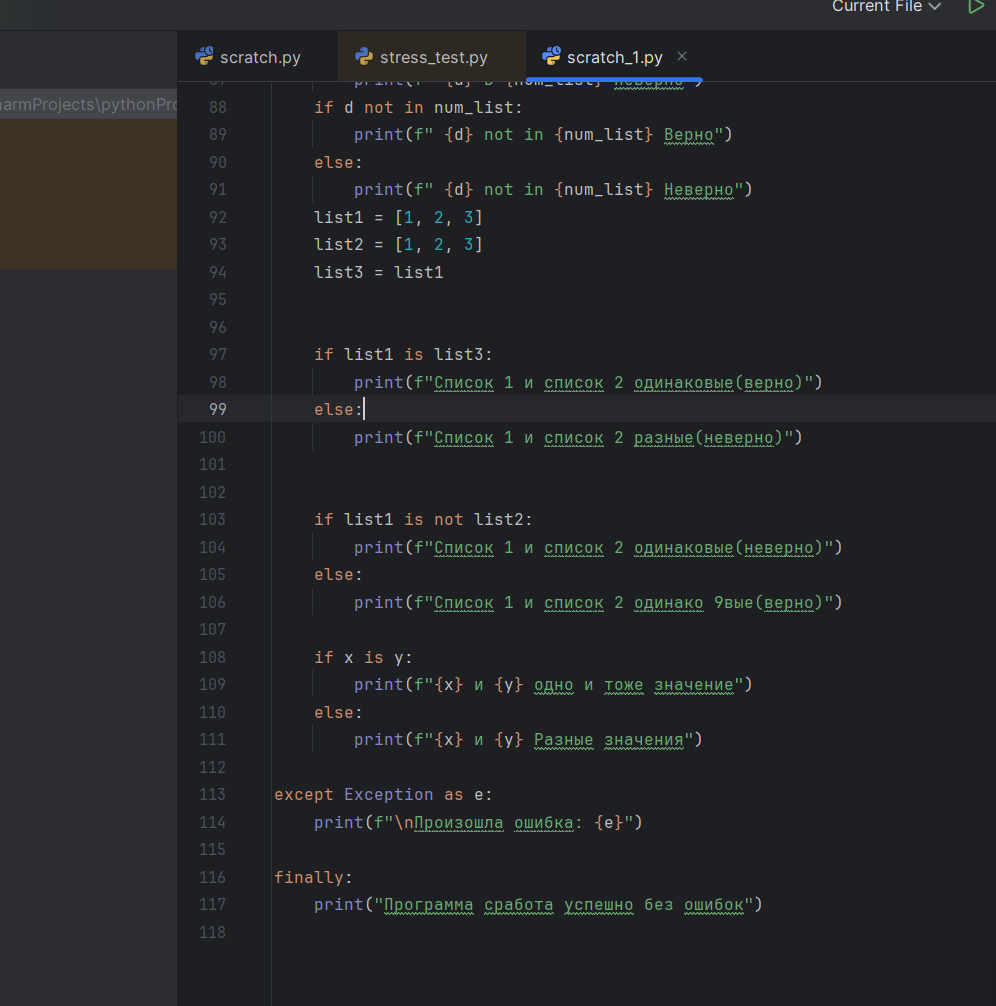


Рисунок 4 Операторы принадлежности

На последнем этапе реализуются операторы принадлежности (проверка, находится ли число в заранее заданном списке) и тождественности (проверка, ссылаются ли две переменные на один объект в памяти).Для обеспечения надежности программа включает обработку исключений с помощью блоков try-except. Это позволяет перехватывать ошибки (например, деление на ноль или ввод нечисловых данных) и выводить информативные сообщения пользователю. После выполнения всех операций программа выводит результат выбранной операции или сообщение об ошибке.

Вывод: В ходе выполнения работы был разработан многофункциональный калькулятор на Python, который успешно реализует все требуемые операции: арифметические, сравнения, логические, побитовые, а также операции принадлежности и тождественности. Программа демонстрирует эффективное использование условных конструкций (if-elif-else) для выбора операций и обработки различных сценариев ввода данных.