Отчёт по практической работе № 3

**Практическое занятие № 3**

**Тема:** составление программ ветвящейся структуры в IDE PyCharm Community.

**Цель:** закрепить усвоенные знания, понятия, алгоритмы, основные принципы составления программ, приобрести навыки составление программ ветвящейся структуры в IDE PyCharm Community.

**Постановка задачи.**

Вариант 6.

1. Смоделировать простейший калькулятор, умеющий выполнять 4 основные арифметические операции.

Вариант 7.

1. Даны три целых числа: A, B, C. Проверить истинность высказывания: «Число B находится между числами A и C».

**Тип алгоритма:** ветвящийся.

**Блок-схема алгоритма 1:**

Вывод t

**Начало**

**Ввод A, B, C**

**Type(A or B or C) != float**

**Обработка исключений**

**Type(A or B or C) = float**

**False**

**A < B < C or A > B > C**

**True**

**Высказывание правдиво**

**Высказывание ложно**

**Конец**

**Текст программы:**

**Задание 1**

            print("Условие: (Вариант 7)")

            print("Даны три целых числа: A, B, C.\n" +

                "Проверить истинность высказывания:\n" +

                "«Число B находится между числами A и C»")

            while True:

                try:

                    a = *float*(input("Введите A: "))

                    b = *float*(input("Введите B: "))

                    c = *float*(input("Введите C: "))

                    if a < b < c or a > b > c:

                        print("Высказывание верно")

                        print(*f*"{a} - {b} - {c}")

                        break

                    else:

                        print("Высказывание ложно")

                        print(*f*"{a} - {b} - {c}")

                        break

                except *ValueError*:

                    print("Некорректные данные")

Протокол работы программы:

1)

Введите A: 16

Введите B: 15

Введите C: 12

Высказывание верно

16 – 15 – 12

Process finished with exit code 0

2)

Введите A: 11

Введите B: 15

Введите C: 12

Высказывание ложно

11 – 15 – 12

Process finished with exit code 0

**Блок-схема алгоритма 2:**

**Начало**

**Ввод A, B, act**

**Type(A and B) != float**

**Обработка исключений**

**Type(A and B) = float**

**act = 4**

**act = 1**

**act**

**act = 3**

**act = 2**

**A \* B**

**A / B**

**A - B**

**A + B**

**Конец**

**Задание 2**

            print("Условие:")

            print("Смоделировать простейший калькулятор,\n" +

            "умеющий выполнять 4 основные арифметические операции\n")

            print("Калькулятор")

            while True:

                try:

                    a = *float*(input("Введите первое число: "))

                    b = *float*(input("Введите второе число: "))

                    act = input("Выберите действие:\n" +

                                " 1 - сумма\n" +

                                " 2 - разность\n" +

                                " 3 - произведение\n" +

                                " 4 – деление\n")

                    if act == "1":

                        print(*f*"Сумма = { a + b }")

                        break

                    if act == "2":

                        print(*f*"Разность = { a - b }")

                        break

                    if act == "3":

                        print(*f*"Произведение = { a \* b }")

                        break

                    if act == "4":

                        print(*f*"Частное = { a / b }")

                        break

                    else:

                        print("Нет такого действия")

                        break

                except *ValueError*:

                    print("Некорректные данные")

Протокол работы программы:

Введите первое число: 16

Введите второе число: 15

Выберите действие:

1 – сумма

2 – разность

3 ­– произведение

4 - деление

2

Разность = 1

Process finished with exit code 0

**Вывод:** в процессе выполнения практического занятия закрепил навыки составления программ ветвящейся структуры в IDE PyCharm Community. Были использованы языковые конструкции while, но в основном if.

Выполнены разработка кода, отладка, тестирование, оптимизация программного кода.

Готовые программные коды выложены на GitHub.