**Практическое занятие № 2**

**Тема:** Составление программ циклической структуры в IDE PyCharm Community.

**Цель:** выработка первичных навыков работы с IDE PyCharm Community, составление программ линейной структуры.

**Постановка задачи.**

Дано два целых числа A и B (A < B). Вывести в порядке возрастания все  
целые числа, расположенные между A и B (включая сами числа A и B), а также количество N этих чисел  
**Тип алгоритма:** линейный.

**Блок-схема алгоритма:**

Вводим числа при помощи

Команды int(input())

При помощи if и try

проверяем введенные значения на условия

Да

Нет

Вывод

результата

**Текст программы:**

# -\*- coding: utf8 -\*-  
# ПЗ 4 вариант 26 задание 1  
# Дано два целых числа A и B(A < B). Вывести в порядке возрастания все  
# целые числа, расположенные между A и B(включая сами числа A и B), а также  
# количество N этих чисел  
  
  
def numbers\_between(a, b):  
 while a >= b:  
 a = int(input("Первое число должно быть меньше второго. Пожалуйста, введите первое число заново: "))  
 b = int(input("Теперь введите второе число: "))  
  
 # Генератор списка для создания списка чисел  
 number\_list\_descending = list(range(b, a - 1, -1))  
  
 print("Числа между A и B в порядке убывания:", \*number\_list\_descending)  
 print("Количество чисел между A и B включительно:", len(number\_list\_descending))  
  
  
a = int(input('Первое число: '))  
b = int(input('Второе число: '))  
  
numbers\_between(a, b)

**Протокол работы программы:**

Первое число: 12

Второе число: 33

Числа между A и B в порядке убывания: 33 32 31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12

Количество чисел между A и B включительно: 22

Process finished with exit code 0

**Вывод:** в процессе выполнения практического занятия выработал навыки составления программ линейной структуры в IDE PyCharm Community В этой программе был использован цикл while для проверки условия a >= b, чтобы гарантировать, что первое число меньше второго. При этом при несоблюдении условия запрашивались новые значения для переменных a и b.

Далее был использован генератор списка для создания списка чисел в порядке убывания между a и b. Метод range использовался с шагом -1 для создания списка чисел в обратном порядке от b до a.

Наконец, выводился список чисел в порядке убывания с помощью print(\*number\_list\_descending), а также количество чисел в этом списке с помощью len(number\_list\_descending).

Программа также запрашивает у пользователя ввод значений a и b для начала выполнения.Выполнены разработка кода, отладка, тестирование, оптимизация кода.

Готовые программные коды выложены на GitHub.