

Практическое занятие №11

Тема: Составление программ для работы с текстовыми файлами.

Цель: Закрепить усвоенные знания, понятия, алгоритмы, основные принципы составления программ, приобрести навыки составления программ с текстовыми файлами в IDE PyCharm Community.

Постановка задачи:

Задача 1. Средствами языка Python сформировать текстовый файл (.txt), содержащий последовательность из целых положительных и отрицательных чисел. Сформировать новый текстовый файл (.txt) следующего вида, предварительно выполнив требуемую обработку элементов:

Исходные данные:

Количество элементов:

Среднее арифметическое элементов:

Последовательность, в которой каждый последующий элемент равен квадрату суммы двух соседних элементов:

Задача 2. Из предложенного текстового файла (text18-18.txt) вывести на экран его содержимое, количество знаков пунктуации в первых четырёх строках. Сформировать новый файл, в который поместить текст в стихотворной форме выведя строки в обратном порядке.

Текст программы:

Блок 1

```
# Создание списка чисел
```

```
numbers = [1, -2, 3, -4, 5]
```

```
# Открытие файла в режиме записи
```

```
with open('output.txt', 'w') as f:
```

```
    for number in numbers:
```

```
        f.write(str(number) + '\n')
```

```
# Вычисление количества элементов, среднего арифметического и последовательности
```

```
count = len(numbers)
```

```
average = sum(numbers) / count
```

```
sequence = [sum(numbers[i:i+2])**2 for i in range(count-1)]
```

```
# Запись результатов обработки в новый текстовый файл
```

```
with open('processed_output.txt', 'w') as f:
```

```
    f.write(f'Количество элементов: {count}\n')
```

```
    f.write(f'Среднее арифметическое элементов: {average}\n')
```

```
    f.write('Последовательность, в которой каждый последующий элемент равен квадрату суммы двух соседних элементов:\n')
```

```
    for number in sequence:
```

```
        f.write(str(number) + '\n')
```

```
# Задача 2: Из предложенного текстового файла (text18-18.txt) вывести на экран его содержимое, количество знаков пунктуации в первых четырёх строках.
```

```
# Сформировать новый файл, в который поместить текст в стихотворной форме выведя строки в обратном порядке.
```

```
# Открытие файла для чтения
```

```
with open('output.txt', 'r') as f:
```

```
    lines = f.readlines()
```

```
# Вывод содержимого файла и подсчет знаков пунктуации в первых четырех строках
```

```
punctuation_count = 0
```

```
for line in lines[:4]:
```

```
    print(line)
```

```
    punctuation_count += sum([1 for char in line if char in '.,!?:;'])
```

```
# Создание файла для записи стихотворной формы текста в обратном порядке
```

```
with open('reversed_poem.txt', 'w') as f:
    # Запись строк в обратном порядке
    for line in reversed(lines):
        f.write(line)

# Вывод количества знаков пунктуации в первых четырех строках
print(f'Количество знаков пунктуации в первых четырех строках: {punctuation_count}')
```

Протокол работы программы 1:

```
1
-2
3
-4
```

Количество знаков пунктуации в первых четырех строках: 0

Process finished with exit code 0

Вывод: В процессе выполнения практического занятия выработала навыки составления программ с использованием текстовых файлов в IDE PyCharm Community. Выполнены разработка кода, отладка, тестирование, оптимизация программного кода. Готовые программные коды выложены на GitHub.