

## 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 №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:

Минимальный элемент:

Квадраты четных элементов:

Сумма квадратов четных элементов:

Среднее арифметическое суммы квадратов четных элементов:

**Тип алгоритма:** Циклический

**Текст программы:**

```
from random import randint
with open('C:\\Users\\Lenovo\\Python
projects\\Projects\\Borzisty\\PZ_11\\file.txt', 'w+') as file:
    # Создаём файл и записываем в него 10 случайных чисел
    file.write(' '.join(str([randint(-100, 100) for i in range(10)])[1:][:1].split(", ")))
with open('C:\\Users\\Lenovo\\Python
projects\\Projects\\Borzisty\\PZ_11\\file.txt', 'r+') as file:
    # Открываем этот же файл на чтение и преобразуем его в int
    sti = list(map(int, file.read().split()))
sq = []
for i in sti:
    if int(i) % 2 == 0:
        i = int(i)**2
        sq.append(i)
with open('C:\\Users\\Lenovo\\Python
projects\\Projects\\Borzisty\\PZ_11\\itog.txt', 'w+') as itog:
    itog.write('Исходные данные: ' + str(sti)[1:][:1] + '\n')
    itog.write('Кол-во элементов: ' + str(len(sti)) + '\n')
    itog.write('Минимальный элемент: ' + str(min(sti)) + '\n')
    itog.write('Квадраты четных элементов: ' + str(sq)[1:][:1] + '\n')
    itog.write('Сумма квадратов четных элементов: ' + str(sum(sq)) + '\n')
    itog.write(f'Среднее арифметическое суммы квадратов четных элементов: {sum(sq)
/ len(sq):.3f}')
```

**Протокол работы программы:**

Process finished with exit code 0

**Постановка задачи №2.**

Из предложенного текстового файла (text18-3.txt) вывести на экран его содержимое, количество знаков пунктуации в первых четырёх строках. Сформировать новый файл, в который поместить текст в стихотворной форме предварительно заменив символы третьей строки их числовыми кодами.

**Тип алгоритма:** Циклический.

**Текст программы:**

```
import string
with open('C:\\Users\\Lenovo\\Python projects\\Projects\\Borzisty\\PZ_11\\text18-3.txt', 'r+', encoding='utf-16') as text:
    tt = text.read()
    print('\n' + tt, '\n')
    text.seek(0)
    four = ' '.join([next(text) for _ in range(4)])
    text.seek(0)
    th = text.readlines()[2]
    three = list(th)
print(f'Кол-во знаков пунктуации в первых 4 строках: {sum([four.count(i) for i in string.punctuation])}')

for i in three[:-1]:
    three[three.index(i)] = ord(i)
new = tt.replace(th, str(three)[1:][:7] + '\n')

with open('C:\\Users\\Lenovo\\Python projects\\Projects\\Borzisty\\PZ_11\\text2.txt', 'w+', encoding='utf-16') as fin:
    fin.write(new)
```

**Протокол работы программы:**

Мы долго молча отступали,  
 Досадно было, боя ждали,  
 Ворчали старики:  
 «Что ж мы? на зимние квартиры?  
 Не смеют, что ли, командиры  
 Чужие изорвать мундиры  
 О русские штыки?»

Кол-во знаков пунктуации в первых 4 строках: 6

Process finished with exit code 0

**Вывод:** в процессе выполнения практического занятия выработал навыки составления программ со строками в IDE PyCharm Community. Были использованы языковые конструкции try, except, input, print, for.

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

Готовые программные коды выложены на GitHub.