

# Практическая №10

## Вариант 2

### Астамирова Елизавета

**Тема:** Составление программ для работы с текстовыми файлами

**Цели практического занятия:** закрепить усвоенные знания, понятия, алгоритмы, основные принципы составления программ, приобрести навыки составления программ с текстовыми файлами в IDE PyCharmCommunity.

#### 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:
# Максимальный элемент:
# Произведение элементов меньших 0 в первой половине:

n = ['75 6 12 -53 47 -3 100 -15'] # Последовательность чисел

# Формирование первого файла с числами
f1 = open('data_1.txt', 'w', encoding='UTF-8')
f1.writelines(n)
f1.close()

# Формирование второго файла с числами
f2 = open('data_2.txt', 'w', encoding='UTF-8')
f2.write('Исходные данные: ')
f2.writelines(n)
f2.close()

# Разбивание строки на элементы и преобразование их в числа
f1 = open('data_1.txt', encoding='UTF-8')
k = f1.read().split()
for i in range(len(k)):
    k[i] = int(k[i])
f1.close()

# Ищем максимальный элемент в файле data_1.txt
f1 = open('data_1.txt', encoding='UTF-8')
max = 0
p = 1
for i in range(len(k)):
    max = max if max > k[i] else k[i]
for i in range(round(len(k)/2)):
    if k[i] < 0:
        p *= k[i]
f1.close()

# Записываем записываем в файл data_2.txt количество элементов и максимальный из них
f2 = open('data_2.txt', 'a', encoding='UTF-8')
f2.write(f'\nКоличество элементов: {len(k)}\nМаксимальный элемент: {max}'
        f'\nПроизведение элементов меньших 0 в первой половине: {p}')
```

**Протокол работы программы:**

75 6 12 -53 47 -3 100 -15

Исходные данные: 75 6 12 -53 47 -3 100 -15

Количество элементов: 8

Максимальный элемент: 100

Произведение элементов меньших 0 в первой половине: -53

## Задание 2.

```
# Из предложенного текстового файла (text18-2.txt) вывести на экран его содержимое, количество знаков препинания.
# Сформировать новый файл, в который поместить текст в стихотворной форме выведя строки в обратном порядке.
z = 0
f1 = open('text18-2.txt', encoding='UTF-8')
print(f1.read())

for i in open('text18-2.txt', encoding='UTF-8'):
    for j in i:
        if j == ',':
            z += 1
        if j == '.':
            z += 1
        if j == '!':
            z += 1
        if j == ':':
            z += 1
        if j == '—':
            z += 1
        if j == '...':
            z += 1

print('\nКоличество знаков препинания: ', z)

# Обратный порядок строк
f1 = open('text18-2.txt', encoding='UTF-8')
N = f1.readlines()
N = N[::-1]
f1.close()

# Создание нового файла со строками в обратном порядке
f2 = open('text18-1.txt', 'w', encoding='UTF-8')
f2.writelines(N)
f2.close()
```

### Протокол работы программы:

— Да, были люди в наше время,  
Не то, что нынешнее племя:  
Богатыри — не вы!  
Плохая им досталась доля:  
Немногие вернулись с поля...  
Не будь на то господня воля,  
Не отдали б Москвы!

Количество знаков препинания: 11

Не отдали б Москвы!  
Не будь на то господня воля,  
Немногие вернулись с поля...  
Плохая им досталась доля:  
Богатыри — не вы!  
Не то, что нынешнее племя:  
— Да, были люди в наше время,

**Вывод:** при выполнении практического задания я закрепила усвоенные знания, понятия, алгоритмы, основные принципы составления программ, приобрести навыки составления программ с текстовыми файлами в IDE PyCharm Community.