

Практическое занятие № 11

Тема: Составление программ для работы с текстовыми файлами в IDE PyCharm Community.

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:

Элементы в обратном порядке:

Сумма элементов последней половины:

2. Из предложенного текстового файла (text18-10.txt) вывести на экран его содержимое, количество букв в верхнем регистре. Сформировать новый файл, в который поместить текст в стихотворной форме предварительно поставив после последней строки автора и название произведения.

Текст программ

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:
```

```
# Элементы в обратном порядке:
```

```
# Сумма элементов последней половины:
```

```
print('17, 9, -3, 1, 2, 10, -4, 7, -2, -1', file=open('file_1.txt', 'w'))# Записываем в  
файл данные
```

```
d = [int(i) for i in open('file_1.txt').read().split(', ')] # Перебор файла запись  
целочисленных в список
```

```
l = open('file_new_1.txt', 'w') # Создаём новый файл
```

```
print('Исходные данные:', open('file_1.txt').read(), file=l) # Выводим
исходные данные
print('Количество элементов:', len(open('file_1.txt').read().split(' ')), '\n', file=l)
# Считаем количество элементов
print('Элементы в обратном порядке:', d[::-1], '\n', file=l) #Выводим
элементы в обратном порядке
print('Сумма элементов последней половины:', sum(d[5:]), file=l) # Считаем
сумму элементов
l.close() # Закрываем .txt файл
```

2.

Из предложенного текстового файла (text18-10.txt) вывести на экран #его
содержимое, количество букв в верхнем регистре.

Сформировать новый файл, в который поместить текст в стихотворной
#форме предварительно поставив после последней строки
автора и название произведения.

```
print(open('text18-10.txt').read()) # Выводим его содержимое
print(open('text18-10.txt').read(), '\n' + 'М. Ю. Лермонтов "Бородино"',
file=open('file18_10.txt', 'w')) # Ставим после последней строки автора и
название
print(sum(map(str.isupper, open('text18-10.txt').read())))) # Считаем и выводим
количество заглавных
```

Протокол работы программы:

1.

В файле file_new_1.txt:

Исходные данные: 17, 9, -3, 1, 2, 10, -4, 7, -2, -1

Количество элементов: 10

Элементы в обратном порядке: [-1, -2, 7, -4, 10, 2, 1, -3, 9, 17]

Сумма элементов последней половины: 10

Process finished with exit code 0

2.

Ну ж был денек! Сквозь дым летучий

Французы двинулись, как тучи,

И всё на наш редут.

Уланы с пестрыми значками,

Драгуны с конскими хвостами,

Все промелькнули перед нами,

Все побывали тут.

8

Process finished with exit code 0

В файле file18_10.txt:

Ну ж был денек! Сквозь дым летучий

Французы двинулись, как тучи,

И всё на наш редут.

Уланы с пестрыми значками,

Драгуны с конскими хвостами,

Все промелькнули перед нами,

Все побывали тут.

М. Ю. Лермонтов "Бородино"

Вывод: в процессе выполнения практического занятия выработала навыки составления программ с текстовыми файлами в IDE PyCharm Community. Были использованы языковые конструкции for i in. ■ Выполнены разработка кода, отладка, тестирование, оптимизация программного кода. Готовые программные коды выложены на GitHub.