

## 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: Среднее арифметическое элементов: Последовательность, в которой каждый последующий элемент равен квадрату суммы двух соседних элементов: 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 выводив строки в обратном порядке.

Тип алгоритма: с ветвлениями.

Текст программы:

1.

```
# 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:

spisok = ['-8 -3 2 4 5 10 17']
f_1 = open('data.txt', 'w')
f_1.writelines(spisok)
f_1.close()

f_2 = open('data_copy.txt', 'w')
f_2.write('Исходные данные: ')
f_2.writelines(spisok)
f_2.write('\n')
f_2.close()

f_1 = open('data.txt')
k = f_1.read()
k = k.split()
for i in range(len(k)):
    k[i] = int(k[i])
f_1.close()

f_2 = open('data_copy.txt', 'a')
kolvo = str(len(k))
f_2.write('Количество элементов: ')
f_2.writelines(kolvo)
f_2.write('\n')
f_2.close()

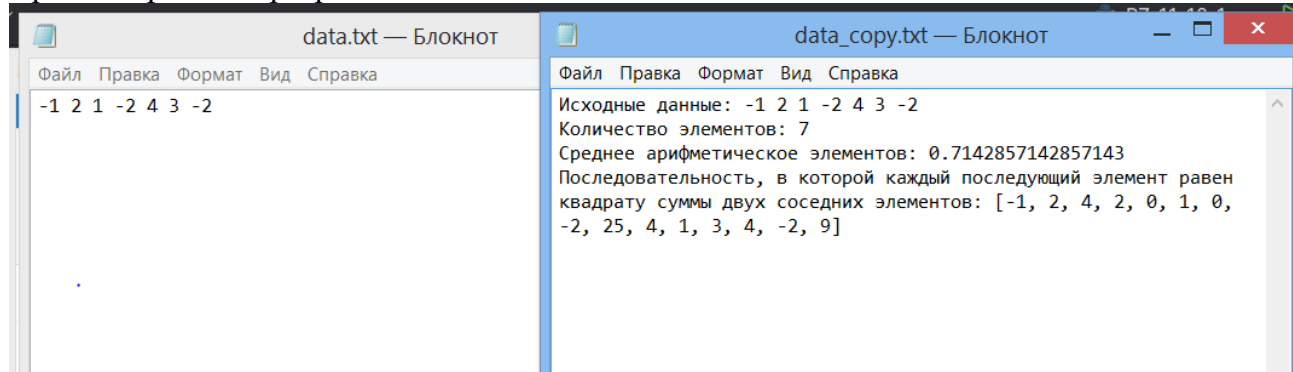
f_2 = open('data_copy.txt', 'a')
three_first = [k[0], k[1], (k[1])**2]
two_last = [k[-1], (k[-2])**2]
n = []
for i in range(1, len(k)-1):
    n.append(((k[i-1] + k[i+1]) ** 2))
k.pop(0)
```

```

k.pop(-1)
spis = []
for i,j in zip(k,n):
    spis.append(i)
    spis.append(j)
itog = three_first+spis+two_last
f_2.write('Последовательность, в которой каждый последующий элемент равен квадрату суммы двух соседних элементов: ')
f_2.writelines(str(itog))
f_2.close()

```

Протокол работы программы:



2.

# 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.

```

with open("text18-18.txt", 'r', encoding="utf-16") as file:
    lines = file.readlines()

count = 0
for line in lines[:4]:
    for i in line:
        if i in '.,?!:;—...':
            count+=1

print("Содержимое файла:")
for line in lines:
    print(line, end="")
print("\n Количество знаков пунктуации в первых четырёх строках:", count)

with open('reverse.txt', 'w', encoding="utf-16") as new_file:
    for line in reversed(lines):
        new_file.write(line)

```

Протокол работы программы:

Содержимое файла:

Вот смерклось. Были все готовы

Заутра бой затеять новый

И до конца стоять...

Вот затрещали барабаны —

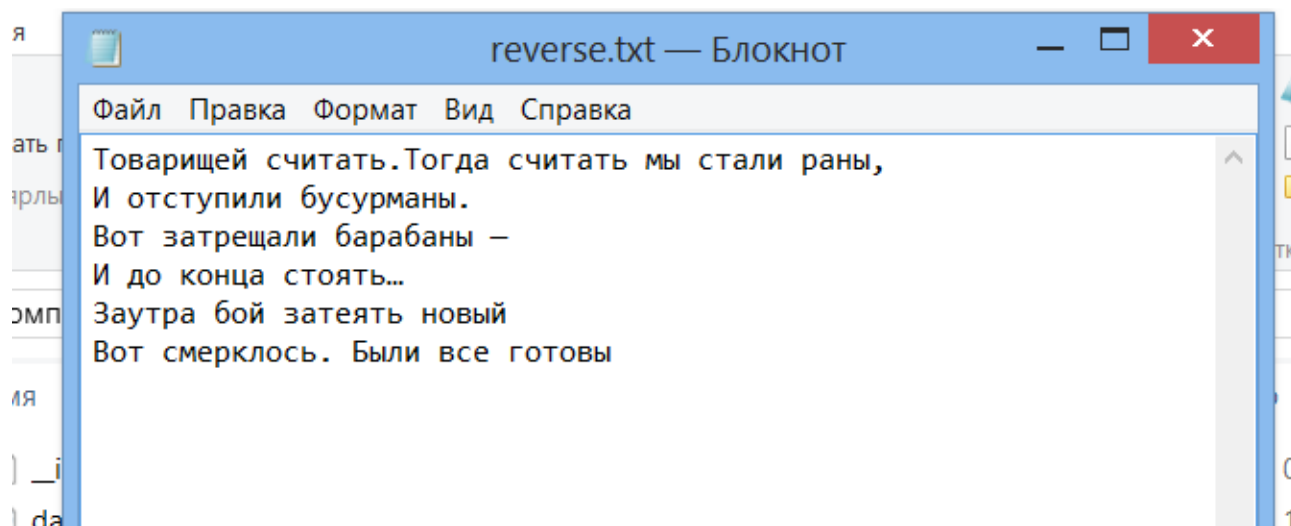
И отступили бусурманы.

Тогда считать мы стали раны,

Товарищей считать.

Количество знаков пунктуации в первых четырёх строках: 3

Process finished with exit code 0



Вывод: в процессе выполнения практического занятия выработала навыки составление программ с текстовыми файлами в IDE PyCharm Community.

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

Готовые программные коды выложены на GitHub.