

Практическое занятие №11-1

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:

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:

- 1) *Исходные данные*
- 2) *Количество элементов*
- 3) *Отрицательные нечетные элементы*
- 4) *Сумма отрицательных нечетных элементов*
- 5) *Среднее арифметическое отрицательных нечетных элементов*

Тип алгоритма: Циклический

Текст программы:

```
nums = ["-99 6 12 -36 20 45 100 -15"]

f1 = open("file_1.txt", "w")
f1.writelines(nums)
f1.close()

# Дублируем список в новый файл file_2.txt
f2 = open("file_2.txt", "w")
f2.write("Исходные данные: ")
f2.writelines(nums)
f2.close()


# Строку преобразуем в числа
f1 = open("file_1.txt")
n = f1.read()
n = n.split()
for i in range(len(n)):
    n[i] = int(n[i])
f1.close()

ii = []
k = 0
for i in range(len(n)):
    if n[i] < 0 and n[i] % 2 != 0:
        ii.append(n[i])
        k += n[i]
gg = k // len(ii)

# Открываем файл для записи
f2 = open("file_2.txt", "a")
f2.write("\n")

# Выводим результат
print(f"Количество элементов: {len(n)}", file=f2)
print("Отрицательные нечётные элементы:", *ii, file=f2)
print(f"Сумма отрицательных нечётных чисел: {k}", file=f2)
print(f"Среднее арифметическое отрицательных нечётных чисел: {gg}", file=f2)
f2.close()
```

Протокол работы программы:

 *file_2.txt – Блокнот

Файл Правка Формат Вид Справка

Исходные данные: -99 6 12 -36 20 45 100 -15

Количество элементов: 8

Отрицательные нечётные элементы: -99 -15

Сумма отрицательных нечётных чисел: -114

Среднее арифметическое отрицательных нечётных чисел: -57

Вывод: В процессе выполнения практического задания выработал навыки работы с файлами в IDE PyCharm. Были использованы языковые конструкции *for*, *open*, *print*. Выполнены разработка кода, отладка, тестирование, оптимизация. Готовые программные коды выложены на GitHub.

Практическое занятие №11-2

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:

Из предложенного текстового файла (text18-1.txt) вывести на экран его содержимое, количество букв в верхнем регистре. Сформировать новый файл, в который поместить текст в стихотворной форме предварительно поставив последнюю строку между первой и второй

Тип алгоритма: Циклический

Текст программы:

```
# Открываем текстовый файл для прочтения его содержимого
f1 = open("text11-1.txt", "r", encoding="utf-16")

# Создаём переменные с которыми будем взаимодействовать
text = ""
uppercase = 0

for l in f1:
    text += l
    uppercase += sum(1 for char in l if char.isupper())

f1.close()
print(text, "\n")
print(f"Количество заглавных букв: {uppercase}")

#####

# Открываем text11-1.txt для обработки содержимого
f2 = open("text11-1.txt", "r", encoding="utf-16")
line = f2.readlines()
f2.close()

# Создаём новый файл text11-2.txt для записи
f2 = open("text11-2.txt", "w", encoding="utf-16")

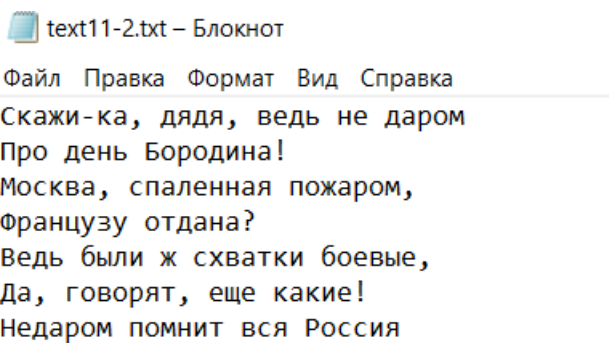
!usage new*
def line_insert():
    line.insert(_index: 1, line[-1])
    del line[-1]
    line[1] += "\n"
line_insert()

# Записываем данные в файл text11-2.txt и закрываем его
f2.writelines(line)
f2.close()
```

Протокол работы программы:

Скажи-ка, дядя, ведь не даром
Москва, спаленная пожаром,
Французу отдана?
Ведь были ж схватки боевые,
Да, говорят, еще какие!
Недаром помнит вся Россия Про
день Бородина!

Количество заглавных букв: 9



Вывод: В процессе выполнения практического задания выработал навыки работы с файлами в IDE PyCharm. Были использованы языковые конструкции *for*, *def*, *open*, *print*. Выполнены разработка кода, отладка, тестирование, оптимизация. Готовые программные коды выложены на GitHub.