

Практическое занятие №11

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 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:

Положительные четные элементы:

Сумма положительных четных элементов:

Среднее арифметическое положительных четных элементов:

Текст программы 1:

```
1  # Средствами языка Python сформировать текстовый файл (.txt), содержащий
2  # последовательность из целых положительных и отрицательных чисел. Сформировать
3  # новый текстовый файл (.txt) следующего вида, предварительно выполнив требуемую
4  # обработку элементов:
5  # Исходные данные:
6  # Количество элементов:
7  # Среднее арифметическое элементов:
8  # Положительные четные элементы:
9  # Сумма положительных четных элементов:
10 # Среднее арифметическое положительных четных элементов:
11
12 print('5, 4, 3, 2, 1, -1, -2, -3, -4, -5', file=open('file58_1.txt', 'w'))
13
14 s, g = [int(i) for i in open('file58_1.txt').read().split(' ')], open('file58_2.txt', 'w', encoding="utf-16")
15
16 a = [i for i in s if i > 0 if i % 2 == 0]
17
18 print("Исходные данные: ", open('file58_1.txt').read(), file=g)
19 print("Количество элементов: ", len(open('file58_1.txt').read().split(' ')), file=g)
20 print("Среднее арифметическое элементов: ", sum(s) / len(s), file=g)
21 print("Положительные четные элементы: ", a, file=g)
22 print("Сумма положительных четных элементов: ", sum(a), file=g)
23 print("Среднее арифметическое положительных четных элементов: ", sum(a) / len(a), file=g)
```

Протокол работы программы 1:

Process finished with exit code 0

```

1 5, 4, 3, 2, 1, -1, -2, -3, -4, -5
2

```

```

1 Исходные данные: 5, 4, 3, 2, 1, -1, -2, -3, -4, -5
2
3 Количество элементов: 10
4 Среднее арифметическое элементов: 0.0
5 Положительные чётные элементы: [4, 2]
6 Сумма положительных чётных элементов: 6
7 Среднее арифметическое положительных чётных элементов: 3.0
8

```

Постановка задачи 2:

Из предложенного текстового файла (text18-7.txt) вывести на экран его содержимое, количество букв в нижнем регистре. Сформировать новый файл, в который поместить текст в стихотворной форме предварительно поставив последнюю строку между второй и третьей.

Текст программы 2:

```

1 # Из предложенного текстового файла (text18-7.txt) вывести на экран его содержимое,
2 # количество букв в нижнем регистре.
3 # Сформировать новый файл, в который поместить текст в стихотворной форме
4 # предварительно поставив последнюю строку между второй и третьей.
5
6 file = open('text18-7.txt', encoding='utf-16')
7 data = file.read()
8 count_letters = sum(map(str.islower, data))
9 print(f"Исходное стихотворение: \n{data}\n")
10 print(f"Количество букв в нижнем регистре: {count_letters}")
11 file.close()
12 file = open('new_text18-7.txt', 'w', encoding='utf-16')
13
14 rows_from_one_to_second = data.splitlines()[2:] # возвращаем строочки, первую и вторую
15 last_row = data.splitlines()[-1]
16 mid_rows = data.splitlines()[2:6]
17 print('\n'.join(rows_from_one_to_second) + '\n' + '\n'.join(last_row) + '\n' + '\n'.join(mid_rows), file=file)
18 file.close()

```

Протокол работы программы 2:

Исходное стихотворение:

Прилег вздремнуть я у лафета,
И слышно было до рассвета,
Как ликовал француз.
Но тих был наш бивак открытый:
Кто кивер чистил весь избитый,
Кто штык точил, ворча сердито,
Кусая длинный ус.

Количество букв в нижнем регистре: 142

Process finished with exit code 0

Вывод:

В ходе выполнения практической работы закрепила усвоенные знания, понятия, алгоритмы, основные принципы составления программ, приобрела навыки составления программ с текстовыми файлами в IDE PyCharm Community.

Были использованы языковые конструкции for, if.

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

Готовые программные коды выложены на GitHub.