

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 два текстовых файла (.txt), содержащих по одной последовательности из целых положительных и отрицательных чисел. Сформировать новый текстовый файл (.txt) следующего вида, предварительно выполнив требуемую обработку элементов: Элементы первого и второго файлов: Количество элементов первого и второго файлов: Количество элементов, общих для двух файлов: Количество четных элементов первого файла: Количество нечетных элементов второго файла:

Текст программы:

```
import random

def subsequence(sign: int = random.choice([1, -1])) -> list:
    """Создаю случайную последовательность"""
    return [random.randint(1, 1000) * sign for i in range(random.randint(175, 200))]

def ints_file(subsequence: list, name: str) -> None:
    """Записываю последовательности в файлы"""
    with open(f"{name}_subsequence.txt", "w") as file:
        file.write(" ".join(map(str, subsequence)))

def get_data_from_file(name: str) -> list:
    with open(f"{name}_subsequence.txt") as file:
        data_file = [int(i) for i in file.read().split(" ")]
        return data_file

ints_file(subsequence(1), "positive")
ints_file(subsequence(-1), "negative")

"""Обрабатываю файлы, как по условию"""
positive_data = get_data_from_file("positive")
negative_data = get_data_from_file("negative")

# Определяю кол-во схожих элементов в 2-х файлах
identical_items, identical_items_without_sign = [], []
for i in positive_data:
    for j in negative_data:
        # Если учитывать минус
        if j == i:
            identical_items.append(j)
        # Если не учитывать минус
        elif j * -1 == i:
            identical_items_without_sign.append(j)

with open("report.txt", "w", encoding="utf-8") as file:
    file.writelines([
        f"Элементы первого файла: {" ".join(map(str, positive_data))}\n",
        f"Элементы второго файла: {" ".join(map(str, negative_data))}\n",
```

```

        f"Количество элементов первого файла: {len(positive_data)}\n",
        f"Количество элементов второго файла: {len(negative_data)}\n",

        f"Количество элементов, общих для двух файлов:
{len(identical_items)}\n"
        f"                Если минус не учитывается:
{len(identical_items_without_sign)}\n",

        f"Количество четных элементов первого файла: {
            len([i for i in positive_data if i % 2 == 0])
        }\n",
        f"Количество нечетных элементов второго файла: {
            len([i for i in positive_data if i % 2 != 0])
        }\n"
    ])
])

```

Протокол работы программы:

Process finished with exit code 0

Студент группы ИС-22 Светличный Н.Д.

Постановка задачи №2.

Из предложенного текстового файла (text18-29.txt) вывести на экран его содержимое, количество символов в тексте. Сформировать новый файл, в который поместить текст в стихотворной форме предварительно поставив последнюю строку между второй и третьей.

Тип алгоритма: Ветвящийся

Текст программы:

```

filename = "text18-19"

with open(f"{filename}.txt", encoding="utf-16") as file:

```

```

data = [i.split("\n")[0] for i in file.readlines()]

print(f"**Содержимое файла:**\n{"\n".join([text for text in data])}")
print(f"**Количество символов в тексте: {sum([len(i) for i in data])}**")

with open(f"processed_{filename}.txt", "w", encoding="utf-16") as file:
    file.writelines([
        "\n".join(data[:2]) + "\n",
        "".join(data[6]) + "\n",
        "\n".join(data[2:6])
    ])

```

Протокол работы программы:

****Содержимое файла:****

Изведал враг в тот день немало,
 Что значит русский бой удалый,
 Наш рукопашный бой!..
 Земля тряслась — как наши груди,
 Смешались в кучу кони, люди,
 И залпы тысячи орудий
 Слились в протяжный вой...

****Количество символов в тексте: 187****

Process finished with exit code 0

Вывод: в процессе выполнения практического занятия №7 выработал навыки составления программ с текстовыми файлами в IDE PyCharm Community.

Выполнены разработка кода, отладка, тестирование, оптимизация программного кода. Готовые программные кода выложены на GitHub.