

ОТЧЕТ

Лабораторная работа №2

Работа с вводом и выводом данных, чтение и запись файлов

Выполнила: Балабанова Н.И.

Группа: ПИН-31Д

Москва 2023

Задание №1. В файле записаны целые числа (создайте самостоятельно). Найти максимальное и минимальное число и записать в другой файл.

```
import os
import re

textFromFile = list()

with open(os.path.dirname(os.path.abspath(__file__)) + '/input.txt', "r") as file:
    content = file.read()
    textFromFile = content.replace('\n', ' ').split(' ')

textFromFileWithoutEmpties = list(filter(None, textFromFile))
regexp = re.compile(r'^-?[0-9]*$')
numbers = [i for i in textFromFileWithoutEmpties if regexp.search(i)]

numbers = list(map(int, numbers))

with open(os.path.dirname(os.path.abspath(__file__)) + '/output.txt', "w") as file:
    if len(numbers) != 0:
        print("max: {0}\nmin: {1}".format(max(numbers), min(numbers)), file=file)
    else:
        print("", file=file)
```

Файл	Содержание	Вывод в output.txt
input.txt	1 2 5 7 8 11 -13 4 6	max: 11 min: -13
input_incorrect.txt	1 2 , 3 4 10dfFF 57 -1 -4j -71 0J --q --1 j-1 j-	max: 57 min: -71
input_incorrect_all.txt	, 10dfFF -4j 0J --q --1 j-1 j-	

Данная программа включает в себя в том числе обработку исключительных ситуаций. В текущем варианте реализации программа выбирает все целые числа из файла в список и на основании полученных данных находит максимальное и минимальное число.

Можно было бы пойти другим путем и, например, при наличии хотя бы одного типа, отличного от `int`, завершать программу, сигнализируя о некорректности вводимых данных.

Задание №2. Считать из файла *input.txt* 10 чисел (числа записаны через пробел). Затем записать их произведение в файл *output.txt*.

```
import os
import re
import numpy

textFromFile = []

with open(os.path.dirname(os.path.abspath(__file__)) + '/input.txt', "r") as file:
    content = file.read()
    textFromFile = content.replace('\n', ' ').split(' ')

regexp = re.compile(r'^-?[0-9].?[0-9]*?$')
numbers = [float(i) for i in textFromFile if regexp.search(i)]

m = 0.0;

if len(numbers) != 0:
    m = numpy.prod(numbers[:10])

with open(os.path.dirname(os.path.abspath(__file__)) + '/output.txt', "w") as file:
    print(m, file=file)
```

Ввод	Выборка чисел	Вывод
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15	1 2 3 4 5 6 7 8 9 10	3628800.0
1 2 3 4 5 r -7 8 9 10 11 12 13 14 15	1 2 3 4 5 -7 8 9 10 11	-6652800.0
1.5 2 g 1 1 0.66	1.5 2 1 1 0.66	1.98
		0.0

Аналогично заданию №1 была предусмотрена обработка ошибок ввода.

Задание №3. Даны два текстовых файла (*input_1.txt*, *input_2.txt*), необходимо записать в файл *output.txt* все слова, которые встречаются в обоих файлах.

```
import os
import string
import numpy

def openFile(fileName):
    with open(os.path.dirname(os.path.abspath(__file__)) + fileName, "r") as file:
        return file.read()

def getAllWords(text):
    t = text.replace('\n', ' ')
    blacklist = string.punctuation + '-'
    new_s = ''.join(char for char in t if char not in blacklist)

    return new_s.split(' ')

def getDataFromFile(file):
    data = openFile(file)
    return getAllWords(data)

listOfWords_1 = getDataFromFile('/input_1.txt')
listOfWords_2 = getDataFromFile('/input_2.txt')

repeated = numpy.intersect1d(listOfWords_1, listOfWords_2)
repStr = ' '.join(repeated)

with open(os.path.dirname(os.path.abspath(__file__)) + '/output.txt', "w") as file:
    print(repStr, file=file)
```

input_1.txt	input_2.txt	output.txt
<p>—порой мне снится, что я кто-то, иду куда-то, знаю цель. я просыпаюсь среди ночи с холодным потом на лице. я душ приму и с самокруткой спущусь в закрытое метро, там закурю и прыгну в утро, где слава богу я никто.</p> <p>—тяну за ниточки обид, бродя по пыльным лабиринтам.</p>	<p>—порой мне снится, что я кто-то, иду куда-то, знаю цель.</p> <p>—сегодня мир так мил, спокоен, что слышен ход далёких вод. и я наедине с собою листаю мысли взад-вперёд.</p> <p>—тяну за ниточки обид, бродя по пыльным лабиринтам. они всегда меня приводят к кривым разбитым зеркалам.</p>	<p>бродя за знаю и иду ктото кудато лабиринтам мне ниточки обид по порой пыльным с снится тяну цель что я</p>

Задание №4. Дан текстовый файл *input.txt*. Определить частоту повторяемости каждой кириллической буквы в тексте, отсортировать в порядке убывания частоты, результат записать в файл *output.txt*. Продемонстрировать работу алгоритма на файлах различной длины

```
import os
import sys

def openFile(fileName):
    with open(os.path.dirname(os.path.abspath(__file__)) + "/" + fileName, "r") as file:
        return file.read()

def writeToFile(fileName, data):
    with open(os.path.dirname(os.path.abspath(__file__)) + "/" + fileName, "w") as file:
        print(data, file=file)

whitelist = "абвгдеёжзийклмнопрстуфхцчшщъыьэюя"

def prepareData(text):
    new_s = ''.join(char for char in text if char in whitelist)
    return new_s

def analyzeData(text):
    data = prepareData(text.lower())
    rate = {}

    for letter in whitelist:
        rate[letter] = data.count(letter)

    rate = sorted(rate.items(), key=lambda x:x[1], reverse=True)

    return rate

def main():
    if len(sys.argv) != 2:
        print("usage: \nlab_2_4.py [filename]")
        return

    fileName = sys.argv[1]

    data = openFile(fileName)
    rate = analyzeData(data)

    writeToFile('output.txt', rate)

main()
```

lab_2_4.py

Файл	Размер	Скрипт	Скорость обработки
input.txt	376 Б	lab_2_4.py	0,02s user 0,01s system 84% cpu 0,037 total
		lab_2_4_read_by_lines.py	0,02s user 0,01s system 85% cpu 0,038 total
[('о', 19), ('т', 12), ('с', 11), ('у', 11), ('а', 10), ('р', 10), ('и', 9), ('н', 8), ('е', 7), ('к', 7), ('м', 7), ('п', 7), ('д', 6), ('я', 5), ('л', 4), ('ы', 4), ('в', 3), ('г', 3), ('з', 3), ('ь', 3), ('ю', 3), ('й', 2), ('ц', 2), ('ч', 2), ('б', 1), ('х', 1), ('ш', 1), ('щ', 1), ('ё', 0), ('ж', 0), ('ф', 0), ('ъ', 0), ('э', 0)]			

Файл	Размер	Скрипт	Скорость обработки
input_a_lot_of_text.txt	71 КБ	lab_2_4.py	0,03s user 0,01s system 86% cpu 0,044 total
		lab_2_4_read_by_lines.py	0,03s user 0,01s system 86% cpu 0,043 total
[('a', 260375), ('o', 235284), ('и', 180927), ('e', 166603), ('н', 144151), ('з', 126353), ('с', 125763), ('т', 119068), ('р', 110949), ('к', 105093), ('п', 104628), ('в', 75936), ('д', 64829), ('л', 60686), ('у', 56746), ('б', 48615), ('я', 47933), ('м', 44784), ('ы', 35434), ('ч', 24094), ('й', 21734), ('ц', 18190), ('ш', 16944), ('г', 12823), ('ь', 12456), ('ф', 10922), ('ю', 6994), ('щ', 4893), ('х', 3316), ('ж', 2788), ('э', 299), ('ё', 49), ('ъ', 15)]			
not_for_share.json	207,2 МБ	lab_2_4.py	6,76s user 0,38s system 99% cpu 7,184 total
		lab_2_4_read_by_lines.py	7,05s user 0,21s system 99% cpu 7,262 total
[('a', 260375), ('o', 235284), ('и', 180927), ('e', 166603), ('н', 144151), ('з', 126353), ('с', 125763), ('т', 119068), ('р', 110949), ('к', 105093), ('п', 104628), ('в', 75936), ('д', 64829), ('л', 60686), ('у', 56746), ('б', 48615), ('я', 47933), ('м', 44784), ('ы', 35434), ('ч', 24094), ('й', 21734), ('ц', 18190), ('ш', 16944), ('г', 12823), ('ь', 12456), ('ф', 10922), ('ю', 6994), ('щ', 4893), ('х', 3316), ('ж', 2788), ('э', 299), ('ё', 49), ('ъ', 15)]			
not_fot_share.txt	1,04 ГБ	lab_2_4.py	33,99s user 4,19s system 93% cpu 40,884 total
		lab_2_4_read_by_lines.py	35,33s user 1,34s system 99% cpu 36,749 total
[('a', 1301875), ('o', 1176420), ('и', 904635), ('e', 833015), ('н', 720755), ('з', 631765), ('с', 628815), ('т', 595340), ('р', 554745), ('к', 525465), ('п', 523140), ('в', 379680), ('д', 324145), ('л', 303430), ('у', 283730), ('б', 243075), ('я', 239665), ('м', 223920), ('ы', 177170), ('ч', 120470), ('й', 108670), ('ц', 90950), ('ш', 84720), ('г', 64115), ('ь', 62280), ('ф', 54610), ('ю', 34970), ('щ', 24465), ('х', 16580), ('ж', 13940), ('э', 1495), ('ё', 245), ('ъ', 75)]			

В скрипт lab_2_4_read_by_lines.py внесены изменения, согласно которым файл считывается не целиком в память, а обрабатывается построчно. С кодом можно ознакомиться в приложении, также как и с содержанием файлов input.txt и input_a_lot_of_text.txt

Файлы not_for_share.* содержат в себе логи с боевого сервиса, приложить нет возможности из-за NDA компании.