



Министерство науки и высшего образования Российской Федерации
Калужский филиал
федерального государственного бюджетного
образовательного учреждения высшего образования
«Московский государственный технический университет имени Н.Э. Баумана
(национальный исследовательский университет)»
(КФ МГТУ им. Н.Э. Баумана)

ФАКУЛЬТЕТ ИУК «Информатика и управление»

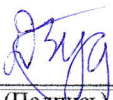
КАФЕДРА ИУК4 «Программное обеспечение ЭВМ, информационные технологии»

ЛАБОРАТОРНАЯ РАБОТА №2

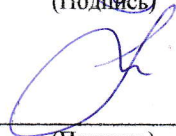
«Работа с файлами и строками на Python»

ДИСЦИПЛИНА: «Перспективные языки программирования»

Выполнил: студент гр. ИУК4-32Б


(Подпись) (Зудин Д.В.)
(Ф.И.О.)

Проверил:


(Подпись) (Пчелинцева Н.И.)
(Ф.И.О.)

Дата сдачи (защиты):

12.10.22

Результаты сдачи (защиты):

- Балльная оценка:

98

- Оценка:

заочно

Калуга, 2022 г.

Цель: приобретение практических навыков, необходимых для разработки задач, решение которых предполагает использование файлов и строк средствами языка Python.

Задачи:

1. Изучить основные методы работы с файлами;
2. Изучить основные методы работы со строками;
3. Изучить типовые алгоритмы решения задач с использованием дополнительных библиотек.

Вариант №5

Формулировка задания №1

Автоматически (не с клавиатуры) создать несколько кортежей, содержащих имя, возраст и номер телефона. Разложить их на переменные и получить из них два списка: список имён и список номеров. Вывести их на экран.

Листинг программы для задания №1

```
from faker import Faker
import random as rand

fake = Faker('ru_RU')
students = [(fake.unique.first_name(), rand.randint(18, 24),
fake.phone_number()) for i in range(5)]
print(*students, sep='\n')
names = [i[0] for i in students]
numbers = [i[2] for i in students]
print(f"Список имен: {names}")
print(f"Список номеров: {numbers}")
```

Результат выполнения программы для задания №1

```
('Агата', 24, '+7 (687) 604-6904')
('Наталья', 24, '83495384454')
('Игорь', 23, '+7 (737) 157-89-89')
('Акулина', 20, '+7 (332) 522-95-34')
('Никита', 19, '8 (162) 793-88-50')
Список имен: ['Агата', 'Наталья', 'Игорь', 'Акулина', 'Никита']
Список номеров: ['+7 (687) 604-6904', '83495384454', '+7 (737) 157-89-89',
'+7 (332) 522-95-34', '8 (162) 793-88-50']
```

Формулировка задания №2

Дан файл f, в котором содержится несколько строк, все слова в них разделены пробелами. Пользователь вводит любую последовательность символов. Вывести на экран все слова, начинающиеся на эту последовательность.

Листинг программы для задания №2

```
import random as rand

# Создание рандомного файла
with open("task_2.txt", "w") as file:
    for k in range(rand.randint(20, 40)):
        string = []
        for i in range(rand.randint(10, 20)):
            word = ''
            for j in range(rand.randint(5, 10)):
                word += chr(rand.randint(ord('a'), ord('z')))
            string.append(word)
        file.write(' '.join(string) + '\n')

# Считывание файла
content = ""
with open("task_2.txt", "r") as file:
    content = ' '.join(file.readlines())
    content.split(" ")
print(*content, sep='')

# Поиск подстроки в итоговой строке
inp = input("Введите строку: ")
res = [i for i in content.split() if i.startswith(inp)]
print(*res)
```

Результат выполнения программы для задания №2

```
wpyxjhcrvf luucwivqto yhfmrjb jkoqtu bbyhfhb vmzvsiduxo pizey ndwai jizkmeur
mihqgrusn righyayyp aiujjey hpsixao ocprl bbqvb
gllfcr slblzotsda lgaiu hscso xncrcclzo kfhsqvqb cittzcytog xkvtaavpyg
ekbkiaosvb uxywnrhpii fzzhvok jdzgh eaefbt nrhdxycj vzdvbv kbwagrsqi
wpecxrlach tktivm dlwmyff
skxvzqxkp oqppmjd ziaob wjfbubhc fdniei jggoymhujr dbbywvbigj zdiyfox
bjayfkoc wjlnbw1 weqtzi cloed arsqf ujqfwq ytxzixzjos jwcaujc lcgccvycij
lkmpoc gzaosrtq wmrlr
```

rlcfsg pvygcb qxniiby wkmqnh sjcjj vtonvw iambz ahrxzvwl ywlpcuy cnzra
fzeefg nltetpc viqxsny pjlxxujt ekmhzm dozazob piidg aguysaksx
ezehgmw uuvgbe tpnhsxm isixt fzykrddis hnyuz qsecmegsrt jftxqoq dhlneqa
twxetox codroqlc cmtntacmdd hhhhpm kjpzcz spgiyk ashpayo aukmor fpvwquq
vmhkn1
dcrkf myrfwcag jtwyp ngshtlers qaxlyenupy luahxajc oxegv fwkfwjon ikaxi
abehdieud
fwhag zrzqji ieimoxcj zvwrpoe fydhrt ukjesgaliu ulqkrt opfczfu bmzcxcv
qcehfczys sxmxjktxur majxai fyctk rkgqtg xdfwmuj hvhparv
rrmbwboiw gqadg cqyymwyijq bafahrc bejclckkq eaxnzdvqd heijmbpn ltqja
rafrvcnrvr oszhsnehy azlghzvbg gtjlvhzn eqkjvez jfohtw fmkahmz edikgewjlt
qommrka
cjxlse wvssinel wmcawe pkehvy hqmrlmwtz dnlmiiiikm deoehapnu ldyuokp
clncpcdmux thbzqs kosnsow inowe dplysyen
haiubnvtki dnuzrznu eipnmhn vyiyz bzukuezn tomigl uofmdaqp wpvrp mctmtzg
amulwdgxh lnzqnur utxcc tlhygf qtqkjgml
uruqmvsg xydjsslzo abvkiab dvhzm zwezjlqbn zyaflvcn ekehjz zyuewktvp
beqrnsfw yddrkupob wrdsqattq
qwfz ohxbiwj brvhdt wcmactwi hmniya qciuu fypxaovkw xfjhgdxlwr qylmd ezbzi
oqupqe gayvdxnpw veaaznekiq
wlsskzpz hhpqsh fqkgyu cmcxsgv wxqnlrlri jlrkpt dbptzdx mfieobtn wjzb
pmetlveihq
xpolncrzoq jmssbfux oebwhjvvu bkcfytn yjcis azhnxhiuhm zmjecjuon fxozj
yaovuffau klqugnx
oipap dvwipprxq xxezdu echmt abiglrh ezotlwyd hdtbtt hgjxp pxzxmhwc xmlqgxs
pswegiwa hgswtc
fljhppmdbj kabkvtxs hxogz doxajsowfr hxxzeumpvy mlzgghu idnzrpxm czlpev
vrdxrlpnwh ldxilxshh bfxayajpj
wobgksbcm hatgdip oialyvyce eudsbq bogmifu auihl fklho odpysil pyfvjirebx
dnjmf azdwtl
mvrqk wnryaaly qkwuavuyy zpilm uoxrcdlkje igivzba dmnga bcmpnwbmgd
bfkupvtnsa vsucrtgrn rlrjjhd
hvvx dayusd exnvw mhcluvsp kbwzi olwgzr yyomji cunygad hrbmt kmvwigizaov
bwqggoxl prsxf
nlymel aqwqqceh ewfpuznxi qvfcqaucro byhwuxqb jslhe hixlnph grvpchj
elzxsqwi gxpwxu iczgd flauxch
lsvca bobcw exhsaruj urzsirebkz fymlgp caocgaun komwn uuogenzhr zckndgr
icldvincb osztpr gbalabd vpltp bflwaq
lzqfuzxh tmawyyii acnjrzv qjncwcr bsjgkqa lszlk ehrxhi nztqrd ayao
czulehluv apehv gzopbl vgmig fohogurzrx ceaiq nskypa dlxs
jvbhwvj qeiad hempjnhhf aozkjlq wspyffc ihxkypn xnuptbgzkh fyzhw uowvui
cosneivcbu dkfndrlx fglojlscl bfcwexf anmgts fxkyn

```

fxquc ksafuuun haaie kdrvedb lpoxhopkq bqeelq ikdemcwwt jesswo ssladvjf
bewnr fgtvboyqf avzgjzzqmi jlsahn zoeqr
evrehnrnd dmykzqz egyqe frvkccnsh tfwrcbqzot lgnrvxo jrhxrds rcdybwrz izqmq1
igezczpxk ivkojdsq vrbhiyi vdmfhcxg bxsnn lefsdph ufkoeswd lqzzel
vhqob nnjgbdje cvgnif hzipjhs qljxlqci mfljll ptjeisbvz ebesdqs xdahnrvm
avfdmon zmzdwsh oagavusc
kyolqh ssxoaupuz wuzcixdk huseeqwdau flgxgbj kzspxl rcypteukr aukhtsddl
conhcbo enbxakr srsfueayvv jtolbmhgv hfilzoi uykikrtb uuenindd rwanvlf
apmwoset fxvlpndoao lrkbieqy
coyxqsb sswuwj pqjdnc najtzjrm lohxxlo cnzkr yayoa yhxtrq mhwmucxv oftpk
pkyfxtju

```

Введите строку: q

```

qxniiby qsecmegsrt qaxlyenupy qcehfczys qommrka qtqkjgml qwffz qciuu qylmd
qkquavuyy qvfcqaucro qjncwwcr qeiad qljxlqci

```

Формулировка задания №3

Дан файл f, в котором находится несколько строк, каждая из которых состоит из имени и номера телефона. Создать на основе этого файла словарь (ключом являются имена).

Листинг программы для задания №3

```

import random as rand
from faker import Faker

fake = Faker("ru_RU")
# Создание случайного файла
with open("task_3.txt", "w") as file:
    for i in range(rand.randint(20, 40)):
        string = []
        name = fake.unique.first_name() + ';'
        number = fake.phone_number()
        string.append(name)
        string.append(number)
        file.write(' '.join(string) + '\n')

# Чтение файла
content = ""
with open("task_3.txt", "r") as file:
    content = ' '.join(file.readlines())
    content = content.split('\n')[:-1]
    content = [x.split(';') for x in content]

```

```

print(content)

# Составление словаря
d = {x.strip(): y.strip() for x, y in content}
print(d)

```

Результат выполнения программы для задания №3

```

[['Фаина', ' +7 (094) 050-95-15'], ['Станисмир', ' +7 (320) 818-1229'], ['
Фадей', ' +7 (664) 106-00-08'], ['Надежда', ' +7 (104) 398-23-38'], ['
Мария', ' 8 142 592 0462'], ['Агата', ' 8 (538) 259-0091'], ['Аггей', '
88432491550'], ['Клавдий', ' +7 (125) 573-3831'], ['Адриан', ' +7 (452)
429-8503'], ['Мечислав', ' +7 (515) 367-9213'], ['Тимур', ' +7 500 635
4183'], ['Ипатий', ' 8 (574) 798-0978'], ['Валентина', ' 8 241 149 5393'],
['Елизавета', ' +7 562 739 8433'], ['Эраст', ' 8 (736) 442-7763'], ['
Никодим', ' 8 (606) 141-3567'], ['Гаврила', ' 8 (355) 639-19-16'], ['Юлий',
' 85131092296'], ['Кондратий', ' +78839195082'], ['Тарас', ' 8 556 820
9642'], ['Еремей', ' 8 414 956 91 45'], ['Нонна', ' +7 (925) 523-1297'], ['
Аникей', ' 8 014 963 97 06'], ['Лев', ' 8 (888) 909-4960'], ['Светозар', '
+7 (008) 631-72-87'], ['Пров', ' 8 (577) 776-86-77']]

{'Фаина': '+7 (094) 050-95-15', 'Станисмир': '+7 (320) 818-1229', 'Фадей': '+7
(664) 106-00-08', 'Надежда': '+7 (104) 398-23-38', 'Мария': '8 142 592 0462',
'Агата': '8 (538) 259-0091', 'Аггей': '88432491550', 'Клавдий': '+7 (125)
573-3831', 'Адриан': '+7 (452) 429-8503', 'Мечислав': '+7 (515) 367-9213',
'Тимур': '+7 500 635 4183', 'Ипатий': '8 (574) 798-0978', 'Валентина': '8 241
149 5393', 'Елизавета': '+7 562 739 8433', 'Эраст': '8 (736) 442-7763',
'Никодим': '8 (606) 141-3567', 'Гаврила': '8 (355) 639-19-16', 'Юлий':
'85131092296', 'Кондратий': '+78839195082', 'Тарас': '8 556 820 9642',
'Еремей': '8 414 956 91 45', 'Нонна': '+7 (925) 523-1297', 'Аникей': '8 014
963 97 06', 'Лев': '8 (888) 909-4960', 'Светозар': '+7 (008) 631-72-87',
'Пров': '8 (577) 776-86-77'}

```

Формулировка дополнительного задания №1

В строке заменить все заглавные буквы строчными.

Листинг программы для дополнительного задания №1

```

string = input("Введите строку: ")
string = string.lower()
print(f"Изменённая строка: {string}")

```

Результат выполнения программы для дополнительного задания №1

Введите строку: Сегодня я купил рамен для Сергея

Изменённая строка: сегодня я купил рамен для сергея

Формулировка дополнительного задания №2

Дан массив целых чисел. Найти максимальный элемент массива и его порядковый номер.

Листинг программы для дополнительного задания №2

```
import random as rand

n = int(input("Введите количество элементов массива: "))
array = [rand.randint(-100, 100) for i in range(n)]
print("Целочисленный массив:\n", array)
print("Максимальный элемент массива: ", max_el := max(array), "[",
array.index(max_el), "]", sep="")
```

Результат выполнения программы для дополнительного задания №2

Введите количество элементов массива: 3

Целочисленный массив:

[-94, 82, 73]

Максимальный элемент массива: 82[1]

Формулировка дополнительного задания №3

Дан одномерный массив целого типа. Получить другой массив, состоящий только из нечётных чисел исходного массива или сообщить, что таких чисел нет. Полученный массив вывести в порядке убывания элементов.

Листинг программы для дополнительного задания №3

```
import random as rand

size = int(input("Введите количество элементов массива: "))
array_1 = [rand.randint(-100, 100) for i in range(size)]
print("Исходный массив: \n", array_1)
array_2 = [i for i in array_1 if i % 2]
print("Конечный массив: \n", sorted(array_2, reverse=True))
```

Результат выполнения программы для дополнительного задания №3

Введите количество элементов массива: 5

Исходный массив:

```
[-6, 5, -49, -39, 77]
```

Конечный массив:

```
[77, 5, -39, -49]
```

Формулировка дополнительного задания №4

Автоматически (не с клавиатуры) создать несколько кортежей, содержащих имя, возраст и номер телефона. Разложить их на переменные и получить из них два списка: список имён и список номеров. Вывести их на экран.

Листинг программы для дополнительного задания №4

```
from faker import Faker
import random as rand

fake = Faker('ru_RU')
students = [(fake.unique.first_name(), rand.randint(18, 24),
fake.phone_number()) for i in range(5)]
print(*students, sep='\n')
names = [i[0] for i in students]
numbers = [i[2] for i in students]
print(f"Список имен: {names}")
print(f"Список номеров: {numbers}")
```

Результат выполнения программы для дополнительного задания №4

```
('Агата', 24, '+7 (687) 604-6904')
('Наталья', 24, '83495384454')
('Игорь', 23, '+7 (737) 157-89-89')
('Акулина', 20, '+7 (332) 522-95-34')
('Никита', 19, '8 (162) 793-88-50')
Список имен: ['Агата', 'Наталья', 'Игорь', 'Акулина', 'Никита']
Список номеров: ['+7 (687) 604-6904', '83495384454', '+7 (737) 157-89-89',
'+7 (332) 522-95-34', '8 (162) 793-88-50']
```

Выводы:

В ходе работы были приобретены практические навыки, необходимых для разработки задач, решение которых предполагает использование файлов и строк средствами языка Python.