|  |  |
| --- | --- |
| **Gerb-BMSTU_01** | **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 | | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ (\_\_\_\_\_\_Зудин Д.В.\_\_\_\_\_)  (Подпись) (Ф.И.О.) |
| Проверил: | | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ (\_\_\_Пчелинцева Н.И.\_\_\_)  (Подпись) (Ф.И.О.) |
| Дата сдачи (защиты):  Результаты сдачи (защиты): | | |
|  | - Балльная оценка:  - Оценка: | |
| 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 yhfmjrb jkoqtu bbyhfhb vmzvsiduxo pizey ndwai jizkmeur miyhqrusn righyayyp aiujjey hpsixao ocprl bbqvb  gllfcr slblzotsda lgaiu hsccso xcnrcclzo kfhskvqb cittzcytog xkvtavppyg ekbkiaosvb uxywnrhpii fzzhvok jdzhg eaefbt nrhdxyxcj vzdvbv kbwagrsqi wpecxrlach tktivm dlwmyff  skxvzqxkp oqppmjd ziaob wcjfubhc fdniei jjgoymhujr dbbywvbigj zdiyfox bjayfkoc wjlnbwl weqtzi cloed arsqf ujqfwq ytxzixzjos jwcaujc lcgccvycij lkmpoc gzaosrtq wmrlr  rlcfsg pvygcb qxniiby wkmqnh sjcjj vtonvw iambz ahrxzvwl ywlpcuy cnzra fzeefg nltetpc viqxpsny pjlxxujt ekmhzm dozazozb piidg aguysaksx  ezehgmw uuvgbe tpnhsmx isixt fzykrddis hnyuz qsecmegsrt jftxqoq dhlneqa twxetox codroqlc cmtntacmdd hhhhpm kjpzkz spgiyk ashpayo aukmor fpvwquq vmhknl  dcrkf myrfwcag jtwyp ngshtlers qaxlyenupy luahxajc oxegv fwkfwjon ikaxi abehdieud  fwhag zrjqji ieimoxcj zvwrpoe fydhrt ukjesgaliu ulqkrt opfczfu bmzcxv qcehfczys sxmxjktxur majxai fyctk rkgqtg xdfwmuj hvhparv  rrmbwboiw gqadg cqyymwyijq bafahrc bejclckkq eaxnzdvqd heijmbpn ltqja rafrvcnrvr oszhsnehy azlghzvgbg gtjlvhzn eqkjvez jfohtw fmkahmz edikgewjlt qommrka  cjxlse wvssinel wmcbwe pkehvy hqmrllmwtz dnlmiiiikm deoehapnu ldyuokp clcncpdmux thbzqs kosnsov inowe dplysyeen  haiubnvtki dnuzrznu eipnmhn vyiyz bzukuezn tomigl uofmdaqp wpvrp mctmtzg amulwdgxh lnzqnur utxcc tlhygf qtqkjgml  uruqmvsq xydjslzeo abvkiab dvhzm zwezjlqbn zyaflvcm ekehjz zyuewktvp beqrnswf yddrkgupob wrdsqattq  qwffz ohxbiwj brvhdgt wcmactwi hmniya qciuu fypxaovkw xfjhgdxlwr qylmd ezzbi oqupqqe gayvdxnpw veaaznekiq  wlsskzpz hhpgsh fqkgyu cmcxsgv wxqniglri jlrfkpt dbptzdxt mfieobtyn wjabz pmetlveihq  xpolncrzoq jmssbfux oebwhjvvu bkcfytn yjcis azhnxhiuhm zmjecjuqon fxozj yaovuffau klqugnx  oipap dvwipprxq xxezdu echmt abiglrh ezotlwyd hdtbtt hgjxp pxzxmhwqc xmlqgxs pswegiwa hgswtc  fljhppmdbj kabkvtxs hxogz doxajsowfr hzxzeumpvy mlzgghu idnzrpxm czlpev vrdxrlpnwh ldxilxshh bfxyajpjj  wobgksbcm hatgdip oialyvyce eudsbq bogmifu auiihl fklho odpysil pyfvjirebx dnjmfm azdwtl  mvrgk wnryaaly qkwuavuyy zpilm uoxrcdlkje igivzvba dmnga bcmpnwbmgd bfkupvtnsa vsucrtgrn rlrjjhd  hvvcx dayusd exnvw mhcluvsp kbzwzi olwgzr yyomji cunyigad hrbdmt kmvwgizaov bwqqgoxl prsxf  nlymel aqwgqceh ewpfpuznxi qvfcqaucro byhwuxqb jslhe hixlznph grvpchj elzxsqwwi gxnpwexu iczgds flauxch  lvsvca bobcw exchsaruj urzsirebkz fymlgp caocgauqn komwn uuogenzhtr zckndgr icldvinccb osztipr gbalabd vpltqp bflwaq  lzqfuzxh tmawyyii acnjrzv qjncwwcr bsjgkqa liszlk ehrxhi nztnqrd aycao czulehluv apehv gzopbl vgmig fohogurzrx ceaiq nskypa dlxys  jvbhvwj qeiad hempjnhhf aozkjlg wspyffc ihxkypn xnuptbgzkh fyzhw uowvui cosneivcbu dkfndrlx fgleojslcl bfcwexf anmgts fxkyn  fxquc ksafuuun haaiie kdrvedb lpoxhopkq bqeelq ikdemcwwt jesswo ssldvjf bewnr fgtvboyqf avzgjzzqmi jlsahn zoeqr  evrehnrnd dmykzqz egyqe frvkccnsh tfwrcbqzot lgnrvxo jrhxrds rcdybwrb izqmql igezczpxk ivkojdsq vrbhiyi vdmfhcxg bxsnn lefsdph ufkoeswd lqzzel  vhqob nnjgbdje cvgnif hzipjhs qljxlqci mfljll ptjeisbvw ebesdqsd xdahnrvzm avfdmon zmzdwsh oagavusc  kyolqh ssxoaupuz wuzcixdk huseeqwdau flgxgbj kzspxl rcyptteukr aukhtsddl conhcbo enbxakr srsfueayvv jtolbmhgvc hfilzoi uykikrtb uuenindd rwfanvlf apmwoset fxvlpndoao lrkbieqy  coyxqsb sswuwj pqjdnc najtzjrm lohzxlo cnzekr yayoa yhxttrq mhuwmucxv oftpk pkyfxtju  Введите строку: q  qxniiby qsecmegsrt qaxlyenupy qcehfczys qommrka qtqkjgml qwffz qciuu qylmd qkwuavuyy 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 рандомного 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')  # Считывание 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'], [' Станимир', ' +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', 'Станимир': '+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. | | |