

# РОССИЙСКИЙ УНИВЕРСИТЕТ ДРУЖБЫ НАРОДОВ

Факультет физико-математических и естественных наук

Кафедра прикладной информатики и теории вероятностей

## ОТЧЕТ

### ПО ЛАБОРАТОРНОЙ РАБОТЕ №3

дисциплина: *Компьютерный практикум по моделированию*

Студент: Журавлев Георгий Иванович

Группа: НФИбД 02-20

МОСКВА

2021\_\_ г.

## Цель работы:

Знакомство с методами работы со строками, списками (одномерные массивы), словарями и множествами

Список сокращений: -

Список обозначений: -

## Выполнение работы

### Задание 1.

Даны два произвольные списка. Удалите из первого списка элементы присутствующие во втором списке.

```
1 list1 = []
2 list2 = []
3 num_of_elements_list1 = int(input('Input a number of elements for list1: '))
4 num_of_elements_list2 = int(input('Input a number of elements for list2: '))
5 print("list1 elements ")
6 for i in range(num_of_elements_list1):
7     k = int(input(f"element { i } of list1 = "))
8     list1.append(k)
9 print("list2 elements ")
10 for i in range(num_of_elements_list2):
11     k = int(input(f"element { i } of list2 = "))
12     list2.append(k)
13 print("so, list 1 =", list1)
14 print("list 2 =", list2)
15 print("Sorting lists, so after sort there shouldn't be any elements from list2 in list1")
16 for i in list2:
17     if i in list1:
18         list1.remove(i)
19     else:
20         pass
21
22 print("so, list 1 =", list1)
23 print("list 2 =", list2)
24
```

1. Запрашиваем ввод размера двух массивов с клавиатуры
2. Далее заходим в цикл `for`, где пользователь вводит данные для двух массивов:  
`for i in range(num_of_elements_list1):`  
    `k = int(input(f"element { i } of list1 = "))`  
    `list1.append(k)`  
`print("list2 elements ")`  
`for i in range(num_of_elements_list2):`

```
k = int(input(f"element { i } of list2 = "))  
list2.append(k)
```

3. Сортируем массивы с помощью цикла for:

```
for i in list2:
```

```
    if i in list1:
```

```
        list1.remove(i)
```

```
    else:
```

```
        pass
```

пример работы программы:

```
Input a number of elements for list1: 5
Input a number of elements for list2: 10
list1 elements
element 0 of list1 = 1
element 1 of list1 = 2
element 2 of list1 = 3
element 3 of list1 = 4
element 4 of list1 = 110
list2 elements
element 0 of list2 = 1
element 1 of list2 = 2
element 2 of list2 = 3
element 3 of list2 = 4
element 4 of list2 = 5
element 5 of list2 = 6
element 6 of list2 = 7
element 7 of list2 = 90
element 8 of list2 = 1
element 9 of list2 = 2
so, list 1 = [1, 2, 3, 4, 110]
list 2 = [1, 2, 3, 4, 5, 6, 7, 90, 1, 2]
Sorting lists, so after sort there shouldn't be any elements from list2 in list1
so, list 1 = [110]
list 2 = [1, 2, 3, 4, 5, 6, 7, 90, 1, 2]
```

## Задание 2.

```

1 month = {'01': 'january', '02': 'february', '03': 'march',
2         '04': 'april', '05': 'may', '06': 'june',
3         '07': 'july', '08': 'august', '09': 'september',
4         '10': 'october', '11': 'november', '12': 'december'}
5
6 day = {'01': 'first', '02': 'second', '03': 'third',
7        '04': 'fourth', '05': 'fifth', '06': 'sixth',
8        '07': 'seventh', '08': 'eighth', '09': 'ninth',
9        '10': 'tenth', '11': 'eleventh', '12': 'twelfth',
10       '13': 'thirteenth', '14': 'fourteenth', '15': 'fifteenth',
11       '16': 'sixteenth', '17': 'seventeenth', '18': 'eighteenth',
12       '19': 'nineteenth', '20': 'twentieth', '21': 'twenty first',
13       '22': 'twenty second', '23': 'twenty', '24': 'twenty fourth',
14       '25': 'twenty fifth', '26': 'twenty sixth', '27': 'twenty seventh',
15       '28': 'twenty eighth', '29': 'twenty ninth', '30': 'thirtieth',
16       '31': 'thirty first'}
17
18 x = input('input date in numeral form with dots(for example
19         • 21.02.2021):').split('.')
20 print("Date: ", month[x[1]], day[x[0]], ', ', x[2], ' year.')

```

1. Создаем 2 массива и присваиваем нужным элементам значения.
2. Просим ввести данные, разделенные точкой(это будет наш массив из 3-ех эл-ов)  
`x = input('input date in numeral form with dots(for example 21.02.2021):').split('.')`
3. Выводим эти данные  
`print("Date: ", month[x[1]], day[x[0]], ', ', x[2], ' year.')`

```

input date in numeral form with dots(for example 21.02.2021):04.05.2021
Date: may fourth , 2021 year.

Process finished with exit code 0

```

пример работы программы:

### Задание 3.

```

1  list1 = []
2  sorted_list = []
3  num_of_elements_list1 = int(input('Input a number of elements for list1: '))
4  print("list1 elements ")
5  for i in range(num_of_elements_list1):
6      k = int(input(f"element { i } of list1 = "))
7      list1.append(k)
8  print("Sorting lists, so after sort there should be only unique elements from
• list1")
9  for i in list1:
10     if list1.count(i) == 1:
11         sorted_list.append(i)
12     else:
13         pass
14  print("so, before sorting = ", list1)
15  print("after sorting= ", sorted_list)

```

1. Просим ввести данные(см. 1 задание)

2. Запускаем цикл «for i in list1», который записывает в новый массив, только уникальные эл-ты

for i in list1:

if list1.count(i) == 1:

sorted\_list.append(i)

else:

pass

пример работы программы:

```

Input a number of elements for list1: 10
list1 elements
element 0 of list1 = 1
element 1 of list1 = 1
element 2 of list1 = 2
element 3 of list1 = 2
element 4 of list1 = 3
element 5 of list1 = 3
element 6 of list1 = 4
element 7 of list1 = 5
element 8 of list1 = 6
element 9 of list1 = 7
Sorting lists, so after sort there should be only unique elements from list1
so, before sorting = [1, 1, 2, 2, 3, 3, 4, 5, 6, 7]
after sorting= [4, 5, 6, 7]

```

#### Задание 4.

**Вариант 1** Дана строка, содержащая русскоязычный текст. Найти количество слов, начинающихся с буквы "е".

```
1 import re
2 Str_1 = input(' введите строку , содержащую русскоязычный текст: ')
3 print("The original list : ", Str_1)
4 out_put = re.findall(r'\be\w+', Str_1)
5 print("Слова начинающиеся с 'е' : ", str(out_put))
6 print("кол-во слов начинающиеся с 'е' : ", len(out_put))
```

1. Используем модуль регулярных выражений «re»
2. Благодаря строке `out_put = re.findall(r'\be\w+', Str_1)`, найдем все слова начинающиеся с «е», выведем и посчитаем их.

пример работы программы:

```
введите строку , содержащую русскоязычный текст: еду приеду уеду еда адаптер
The original list : еду приеду уеду еда адаптер дедушка фикус европа
Слова начинающиеся с 'е' : ['еду', 'еда', 'европа']
кол-во слов начинающиеся с 'е' : 3

Process finished with exit code 0
|
```

#### Задание 5.1.

Дан одномерный массив, состоящий из N целочисленных элементов. Ввести массив с клавиатуры. Найти максимальный элемент. Вывести массив на экран в обратном порядке.

```
2 arr_1 = []
3 num_of_elements_arr_1 = int(input('Input a number of elements for array: '))
4 for i in range(num_of_elements_arr_1):
5     k = int(input(f"element { i } of an array = "))
6     arr_1.append(k)
7 print("the largest number in array is ", max(arr_1))
8 print("normal array", arr_1)
9 print("reversed array", list(reversed(arr_1)))
10
11
```

1. Вводим данные с клавиатуры( как и прежде)(см. задание 1 и 3)
2. Выведем максимальный элемент массива с помощью функции `max()`.

3. С помощью вызова `list(reversed(arr_1))` выведем на экран реверсированный массив.

пример работы программы:

```
/usr/bin/python3.8 /home/g_zhuravlev/python-projects/1s
Input a number of elements for array: 5
element 0 of an array = 1
element 1 of an array = 2
element 2 of an array = 3
element 3 of an array = 4
element 4 of an array = 5
the largest number in array is 5
normal array [1, 2, 3, 4, 5]
reversed array [5, 4, 3, 2, 1]

Process finished with exit code 0
```

### Задание 5.2.

**В массиве действительных чисел все нулевые элементы заменить на среднее арифметическое всех элементов массива.**

```
1  arr_1 = []
2  num_of_elements_arr_1 = int(input('Input a number of elements for array: '))
3  for i in range(num_of_elements_arr_1):
4      k = float(input(f"element { i } of an array = "))
5      arr_1.append(k)
6  sr_ar = 0.0
7  print("before: ", arr_1)
8  for j in range(num_of_elements_arr_1):
9      sr_ar += arr_1[j]
10 sr_ar /= num_of_elements_arr_1
11 for i in range (num_of_elements_arr_1):
12     if(arr_1[i] == 0):
13         arr_1[i] = sr_ar
14     else:
15         pass
16 print("sr_ar = ", sr_ar)
17 print("after: ", arr_1)
```

1. Вводим данные с клавиатуры( как и прежде)(см. Задание 1,3, 5.2)

2. Посчитаем среднее арифметическое с помощью итераций

```
for j in range(num_of_elements_arr_1):
```



```
sr_ar += arr_1[j]
```

```
sr_ar /= num_of_elements_arr_1
```

3. С помощью итераций заменим все эл-ты массива, которые = 0, на среднее арифметическое.

```
for i in range (num_of_elements_arr_1):
```

```
    if(arr_1[i] == 0):
```

```
        arr_1[i] = sr_ar
```

```
    else:
```

```
        pass
```

пример работы программы:

```
/usr/bin/python3.8 /home/g_zhuravlev/python-projects/1st.project
Input a number of elements for array: 10
element 0 of an array = 0
element 1 of an array = 0
element 2 of an array = 0
element 3 of an array = 0
element 4 of an array = 2.5
element 5 of an array = 2.4
element 6 of an array = 123
element 7 of an array = 23
element 8 of an array = 0
element 9 of an array = 3
before: [0.0, 0.0, 0.0, 0.0, 2.5, 2.4, 123.0, 23.0, 0.0, 3.0]
sr_ar = 15.39
after: [15.39, 15.39, 15.39, 15.39, 2.5, 2.4, 123.0, 23.0, 15.39, 3.0]
Process finished with exit code 0
```

можно работать с массивами, какие у них есть методы; некоторые новые методы, связанные с массивами и не только. Научился применять и совмещать их.

### Контрольные вопросы:

1. int(integer); float(floating point); str(string); bool(boolean), may be long(long integer)

2. т. к. чаще всего используются и из них можно создавать новые типы данных

3. К immutable типам относятся: int, float, complex, bool, tuple, str и frozen set.

Неизменяемость типа данных означает невозможность изменения содержимого, без изменения объекта.

4. Списки, кортежи и строки - последовательности.

Основные возможности – это проверка принадлежности (т.е. выражения “in” и “not in”) и оператор индексирования, позволяющий получить напрямую некоторый элемент последовательности.

5. 1. с помощью модуля re и команды  $\rightarrow$  `out_put = re.findall(r'\p{w}+', S)`

2. с помощью команды `x = S.split(«, »)` и вывода `S [x[1]]`