Федеральное государственное образовательное бюджетное

учреждение высшего образования

«**Финансовый университет**

**при Правительстве Российской Федерации»**

**(Финансовый университет)**

**Департамент анализа данных, принятия решений и финансовых технологий**

**Отчёт по семинару №6**

**Выполнил:**

Студент Малкеров Г.А.

Группы ЗБ-ПИ1-2

**Проверил:**

Петросов Д.А.

**Москва 2020**

**Задания №№1-11:**

№1 создайте коллекцию на свободную тему и выполните:

№2 подсчет количества членов коллекции с помощью функции len()

№3 проверку принадлежности элемента данной коллекции c помощью оператора in

№4 выполните поиск подстроки

№5 обход коллекции с применением оператора цикла

№6 найдите максимальный, минимальный элементы коллекции и сумму элементов

№7 найдите количество определённого пользователем элемента коллекции

№8 выполните конвертацию типа созданной вами коллекции

№9 выполните сортировку элементов коллекции

№10 реализовать любые два практических задания из темы словари с применением коллекции

№11 реализовать любые два задания из практической работы по спискам и кортежам с применением коллекций

Программный код на языке Python для заданий №1-11:

import collections as c

import traceback

import random

import sys

curCollection = c.Counter("test message")

unicNames = [

"Туров Богдан Валериевич",

"Шухевич Виктор Брониславович",

"Майборода Степан Артёмович",

"Чумак Цефас Алексеевич",

"Молчанов Дмитрий Фёдорович",

"Исаков Родион Эдуардович",

"Медведев Конрад Васильевич",

"Юдин Леонард Иванович",

"Таранец Юлиан Романович",

"Капустин Устин Романович",

"Макаров Ян Григорьевич",

"Овчаренко Юрий Вадимович",

"Федункив Ярослав Владимирович",

"Дроздов Шарль Леонидович",

"Коцюбинский Устин Григорьевич",

"Худобяк Геннадий Сергеевич",

"Поляков Йошка Станиславович",

"Миклашевский Цефас Леонидович",

"Кравчук Ждан Сергеевич",

"Моисеенко Любомир Михайлович",

"Павленко Зуфар Львович",

"Миклашевский Чарльз Максимович",

"Лазарев Орест Леонидович",

"Александров Яромир Данилович",

"Михайлов Жерар Григорьевич",

"Киселёв Шамиль Дмитриевич",

"Батейко Емельян Станиславович",

"Васильев Савва Вадимович",

"Многогрешный Чеслав Станиславович",

"Андреев Карен Михайлович",

"Мартынов Клаус Леонидович",

"Захаров Милан Васильевич",

"Котов Ждан Богданович",

"Кондратьев Огюст Дмитриевич",

"Савин Пётр Иванович",

"Зайцев Николай Виталиевич",

"Пономаренко Цицерон Васильевич",

"Павленко Давид Романович",

"Веселов Леон Виталиевич",

"Фомин Виль Борисович",

"Анисимов Роберт Викторович",

"Третьяков Максим Эдуардович",

"Некрасов Шарль Львович",

"Рябов Чеслав Петрович",

"Буров Пётр Валериевич",

"Фомичёв Харитон Богданович",

"Савенко Шамиль Викторович",

"Казаков Георгий Данилович",

"Виноградов Валериан Сергеевич",

"Мухин Устин Васильевич",

]

intro = """

1. Создать новую коллекцию

2. Подсчитать кол-во членов коллекции с помощью функции len()

3. Проверить принадлежность элемента к коллекции с помощью оператора in

4. Выполнить поиск подстроки

5. Выполнить обход коллекции с применением оператора цикла

6. Найти максимальный, минимальный элементы коллекции, а также сумму всех элементов

7. Найти кол-во определённого пользователем элемента коллекции

8. Выполнить конвертацию типа коллекции

9. Выполнить сортировку элементов коллекции

0. Завершить программу

"""

def InputSomeDataInt():

try:

return [int(x) for x in input("Введите список чисел через пробел ").split(" ")]

except:

print("Ошибка при работе программы ", traceback.format\_exc())

return

def Task1():

""" Создаёт новую коллекцию """

global curCollection

newCollection = input("Введите новые элементы коллекции:\n")

while True:

ans = input("1 - Разделить элем. через пробел\n2 - Не разделять\n:")

if ans == "1":

curCollection = c.Counter(newCollection.split(" "))

break

elif ans == "2":

curCollection = c.Counter(newCollection)

break

else:

print("Неверная команда!\n")

print("Новая коллекция создана!")

def Task2():

print(f"Количество элементов в коллекции: {len(curCollection)}")

def Task3():

element = input("Введите элем., который хотите проверить: ")

print("Элемент {0} коллекции".format(

"принадлежит" if element in curCollection else "не принадлежит"))

def Task4():

find\_counter = 0

substr = input("Введите подстроку: ")

for key in curCollection.keys():

if substr in key:

print(f"Найдено совпадение в ключе '{key}'")

find\_counter += 1

if not find\_counter:

print("Совпадения не найдены!")

def Task5():

print("Элементы коллекции:")

print("Ключ\tЗначение")

for key, value in curCollection.items():

print(f"{key}\t{value}")

def Task6():

print(f"Максимальный элемент: {max(curCollection.values())}")

print(f"Минимальный элемент: {min(curCollection.values())}")

print(f"Сумма элементов: {sum(curCollection.values())}")

def Task7():

key = input("Введите желаемый элем.: ")

print(f"Элемент '{key}' содержится в коллекции {curCollection[key]} раз")

def Task8():

# В демонстрационных целях конвертируется только в пределах функции

print(f"Изначальная коллекция:\n{curCollection}")

print(f"Конвертация в обычный список:\n{list(curCollection)}")

print(f"Конвертация в обычный кортеж:\n{tuple(curCollection)}")

print(f"Конвертация в обычный словарь:\n{dict(curCollection)}")

def Task9():

print("Сортировка по ключу:")

print("Отсортированная коллекция", sorted(curCollection.keys()))

print("Отсортированная коллекция в обратном порядке",

sorted(curCollection.keys(), reverse=True))

print("Сортировка по значению:")

print("Отсортированная коллекция", sorted(curCollection.values()))

print("Отсортированная коллекция в обратном порядке",

sorted(curCollection.values(), reverse=True))

print("Сортировка по паре ключ-значение:")

print("Отсортированная коллекция", sorted(curCollection.items()))

print("Отсортированная коллекция в обратном порядке",

sorted(curCollection.items(), reverse=True))

def Task10():

tasks = {

1: Task10PartOne,

2: Task10PartTwo}

typeOfWork = int(input("Введите номер задания 1-2\n:"))

if(typeOfWork in tasks):

tasks[typeOfWork]()

else:

print("Задание не найдено")

def Task11():

tasks = {

1: Task11PartOne,

2: Task11PartTwo}

typeOfWork = int(input("Введите номер задания 1-2\n:"))

if(typeOfWork in tasks):

tasks[typeOfWork]()

else:

print("Задание не найдено")

def Task10PartOne():

n = unicNames

d = dict()

for i in range(0, len(n), 2):

d["адрес №" + str(i)] = [n[i], n[i + 1]]

print(f"Итого для поиска\n{d}")

s = input("Введите адрес для поиска:")

if s in d:

print("Адрес найден!\nПроживающие по адресу:")

[print(x) for x in d[s]]

else:

print("Данные не найдены!")

def Task10PartTwo():

d = dict()

for e in unicNames:

key = "+7" + str(random.randint(10 \*\* (10- 1), (10 \*\* 10) - 1))

d[key] = e

print(f"Итого для поиска\n{d}")

inputInfo = input("Введите номер телефона для поиска:")

if inputInfo in d:

print("Значение для телефона " + inputInfo + " : " + d[inputInfo])

else:

print("Введенные данные НЕ были найдены")

def Task11PartOne():

n = int(input("Введите N "))

m = int(input("Введите M "))

h = int(input("Введите Н "))

matrix = [[random.randrange(0, 10) for y in range(m)] for x in range(n)]

print("Сгенерированная матрица ")

for im in range(n):

print(matrix[im])

for col in range(m):

inf = [row[col] for row in matrix]

if(h in inf):

print("В столбце {0} найдено число {1}.".format(col+1, h))

def Task11PartTwo():

inputInfo = InputSomeDataInt()

if inputInfo == inputInfo[::-1]:

print("Cписок является симметричным")

else:

print("Cписок является НЕ симметричным")

def KillProgram(): return sys.exit(0)

def Main():

print(intro)

while True:

print(f"Текущая коллекция: {curCollection}")

try:

tasks = {

1: Task1,

2: Task2,

3: Task3,

4: Task4,

5: Task5,

6: Task6,

7: Task7,

8: Task8,

9: Task9,

10: Task10,

11: Task11,

0: KillProgram

}

task\_for\_work = int(input("Введите номер задания: "))

if task\_for\_work in tasks:

tasks[task\_for\_work]()

else:

print("Задание не найдено!")

except Exception:

print("Ошибка при работе программы", traceback.format\_exc())

print()

if \_\_name\_\_ == "\_\_main\_\_":

Main()

**Пример работы программного кода:**

Пример работы программного кода из заданий №№1-6 представлен на рисунке 1.

Пример работы программного кода из заданий №№7-9 представлен на рисунке 2.

Пример работы программного кода для задания №10 представлен на рисунке 3.

Пример работы программного кода для задания №11 представлен на рисунке 4.

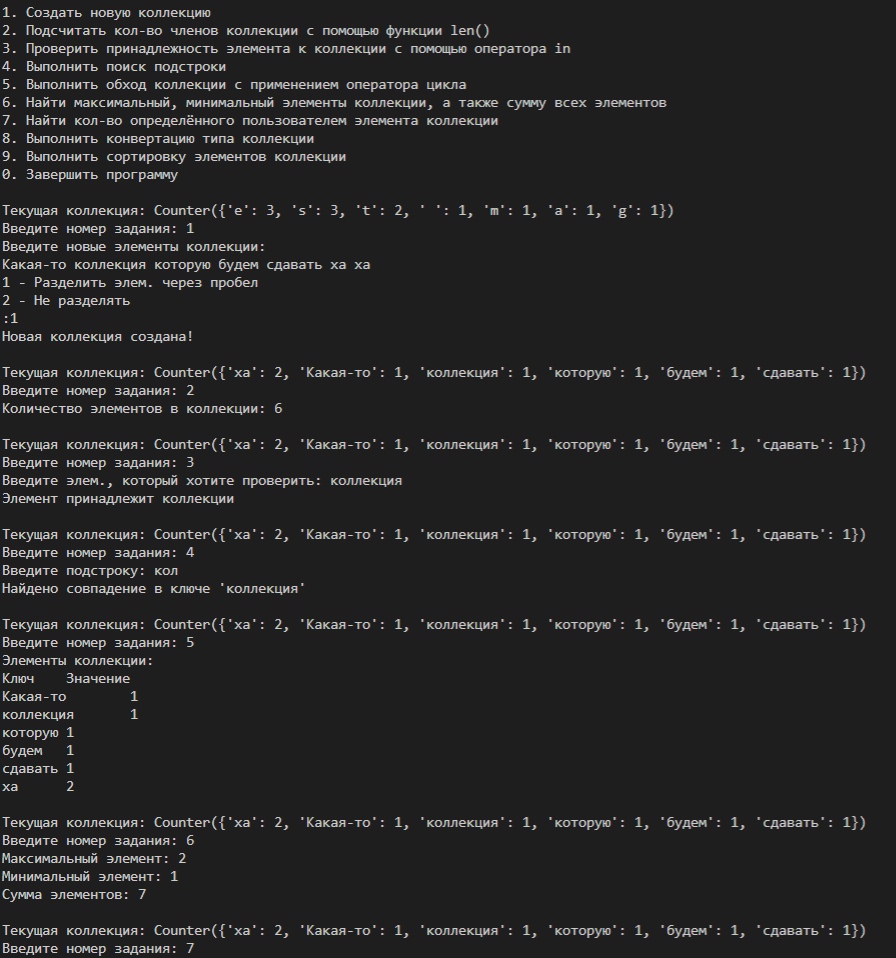


Рисунок 1 – Пример работы программного кода из задания №№1-6

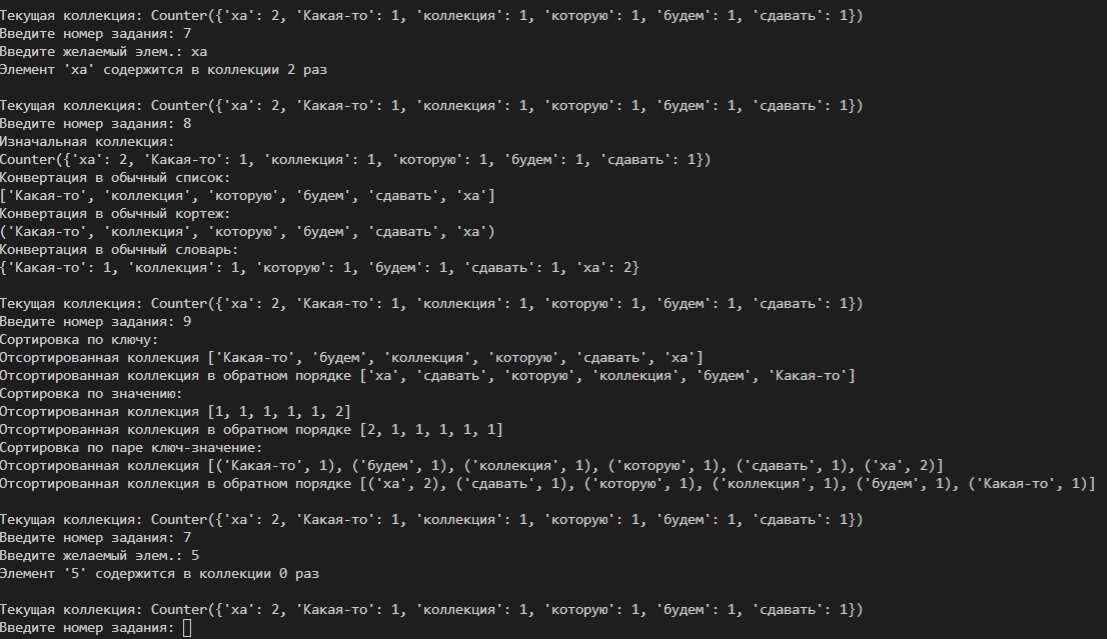


Рисунок 2 – Пример работы программного кода из задания №№7-9

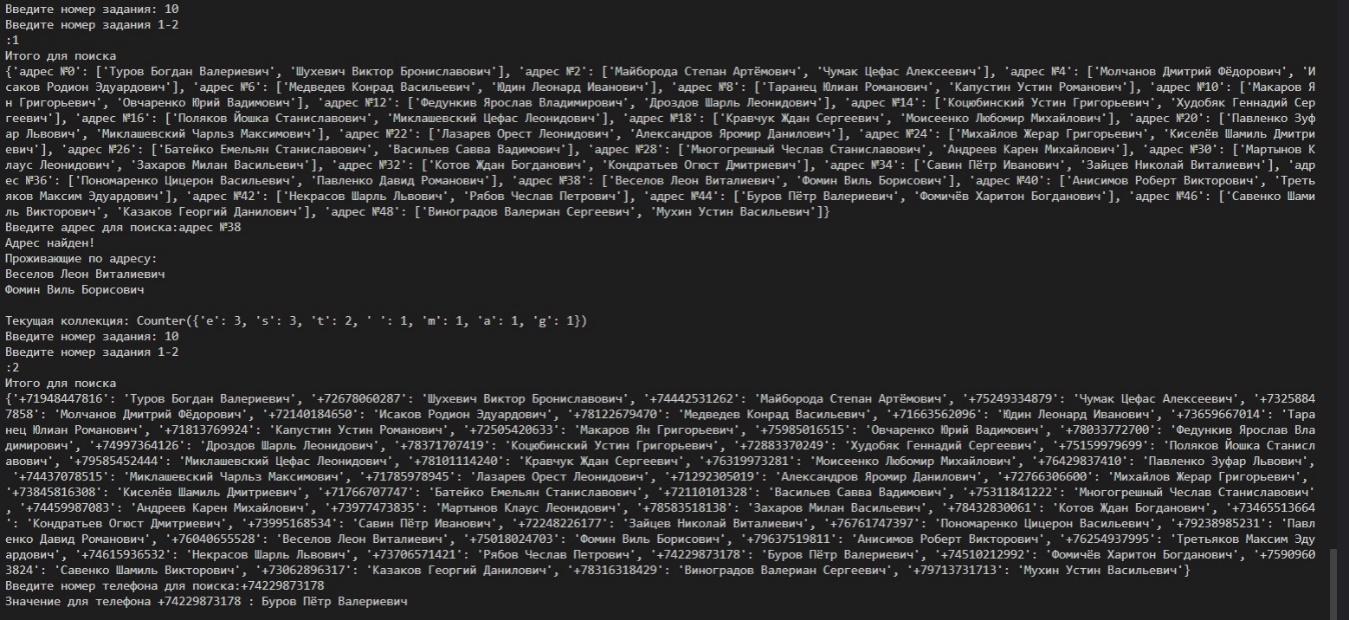


Рисунок 3 – Пример работы программного кода из задания №10

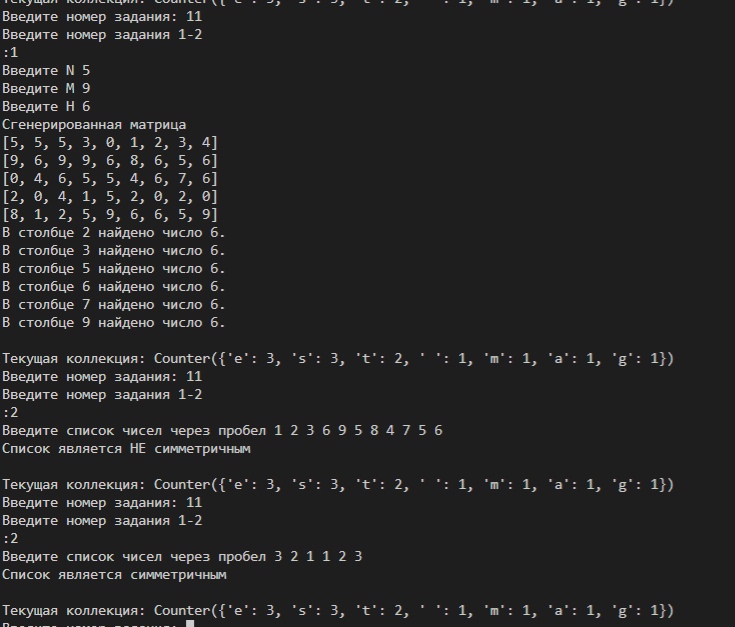


Рисунок 4 – Пример работы программного кода из задания №11

**Вывод:** В ходе выполнения данной практической работы были получены навыки работы с коллекциями в языке программирования Python