**Федеральное агентство связи**

**Ордена Трудового Красного Знамени**

**Федеральное государственное бюджетное образовательное учреждение высшего образования**

**«Московский технический университет связи и информатики»**

Кафедра Математической кибернетики и информационных технологий

**Отчет по лабораторной работе №3**

по дисциплине «Теория языков программирования»

Выполнил:

студент группы БСТ1801

Францев А.Р.

Москва 2019

1. Цель работы: освоение базовых навыков работы с функциями и файлами в Python.
2. Задание для освоения базовых навыков работы с функциями и файлами в Python.
   1. Первое задание:

Посчитайте количество файлов в заданной пользователем директории (папке) и вывести на экран.

* 1. Второе задание:

Выведите информацию о товарах в файле products.txt, отсортировав их по цене.

* 1. Третье задание:

Добавить ко второму заданию пользовательский интерфейс по уменьшению количества всех товаров указанных пользователем номеров на введенное пользователем число.

* 1. Четвертое задание:

Добавьте к пользовательскому интерфейсу из задачи №3 возможность сохранения новых данных обратно в файл. Предусмотрите возможность сохранения данных в отсортированном виде в отдельный файл по выбору пользователя.

1. Ход работы
   1. Код программы

import os

file = open("products.txt", "r")

sort = []

for line in file:

line = line.strip()

line = line.split(";")

sort.append(line)

max = int(sort[len(sort)-1][0])

def Work():

global availability

answer = input('Вы хотите продолжить? Y/N \n')

if answer.lower() == 'n':

availability = False

def Find():

path = input('Введите директорию: ')

count = str(len(next(os.walk(path))))

print(count)

def Sort():

global sort

new\_sort = sort

for j in range(0, len(new\_sort)):

for i in range(0, len(new\_sort)-1):

if int(new\_sort[i][3]) > int(new\_sort[i+1][3]):

stroka = new\_sort[i]

new\_sort[i] = new\_sort[i+1]

new\_sort[i+1] = stroka

return new\_sort

def Change\_stock():

global max

new\_sort = Sort()

id = []

print('Введите id товаров, у которых хотите изменить количество по одному.\n'

'stop - остановка ввода id товаров. \n'

'Максимальный id = ', max)

work1 = True

while work1:

add = input('ID: ')

if add.isdigit():

add = int(add)

if add <= max:

id.append(add)

else:

print('Максимальный ID = ', max, ', Вы ввели: ', add)

elif add.lower() == 'stop':

work1 = False

else:

print('Пожалуйста, вводите числа, а не буквы и спец. символы.')

count = int(input('Введите на сколько хотите уменьшить количество товара,'

'если число будет превышать количество, то количество станет равным 0: '))

for i in range(0, len(new\_sort)):

for j in range(0, len(id)):

if int(new\_sort[i][0]) == int(id[j]):

if int(new\_sort[i][3]) - count >= 0:

new\_sort[i][3] = int(new\_sort[i][3]) - count

else:

new\_sort[i][3] = 0

return new\_sort

def SaveNew():

new\_sort = Change\_stock()

work = True

while work:

answer = input('Вы желаете сохранить результат как новый файл? Y/N \n')

if answer.lower() == 'y':

file\_name = input('Тогда, введите название файла: ')

new\_file = open(file\_name, 'w')

for i in range(0, len(new\_sort)):

for j in range(0, len(new\_sort[i])):

if 0 <= j <= 2:

new\_file.write(str(new\_sort[i][j]) + ";")

else:

new\_file.write(str(new\_sort[i][j]))

new\_file.write('\n')

if answer.lower() == 'n':

new\_file = open('products.txt', 'r+')

for i in range(0, len(new\_sort)):

for j in range(0, len(new\_sort[i])):

if 0 <= j <= 2:

new\_file.write(str(new\_sort[i][j]) + ";")

else:

new\_file.write(str(new\_sort[i][j]))

new\_file.write('\n')

work = False

new\_file.close()

else:

work = False

print("0 - выйти из программы \n1 - Функция номер 1 \n2 - Функция номер 2\n3 - Функция номер 3\n4 - Функция номер 4")

availability = True

while availability:

command = input('Введите команду: ')

if command == "0":

something = input('Вы действительно хотите выйти из программы? Y/N \n')

if something.lower() == 'y':

availability = False

elif something.lower() != 'n':

print('Неизвестная команда!')

if command == "1":

Find()

if command == "2":

print(Sort())

if command == "3":

print(Change\_stock())

if command == "4":

SaveNew()

if not command.isdigit():

print('Неизвестная команда')

if not 'something' in globals() or not 'something' in locals():

Work()

Скриншот выполнения первой подпрограммы представлен на рисунке 1

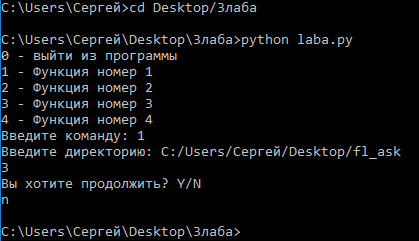


Рисунок 1 - выполнение первой подпрограммы

Скриншот выполнения второй подпрограммы представлен на рисунке 2

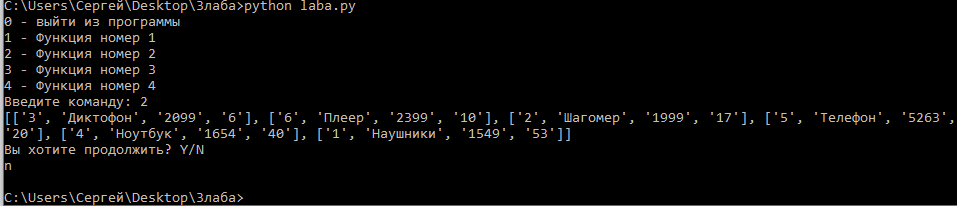


Рисунок 2 - скриншот выполнения второй подпрограммы

Скриншот выполнения третьей под программы представлен на рисунке 3.

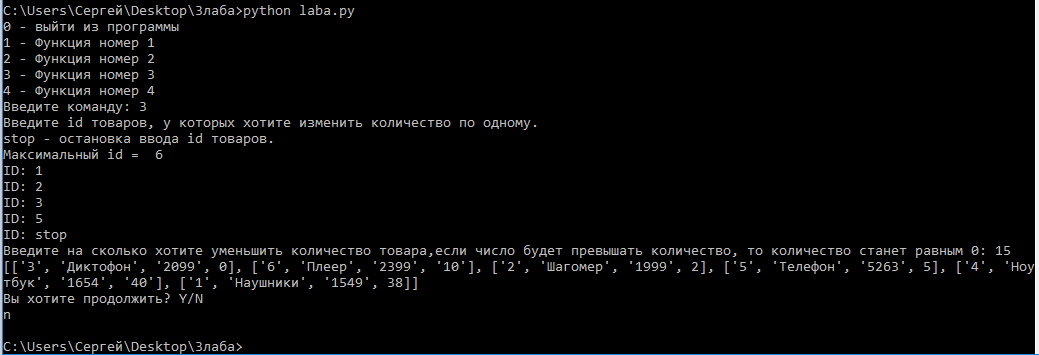


Рисунок 3 - результат выполнения третьей подпрограммы

Скриншот содержимого исходного файла представлен на рисунке 4.

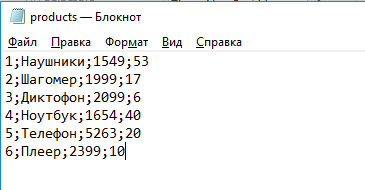


Рисунок 4 - содержимое исходного файла

Скриншот выполнения 4 подпрограммы представлен на 5 рисунке.

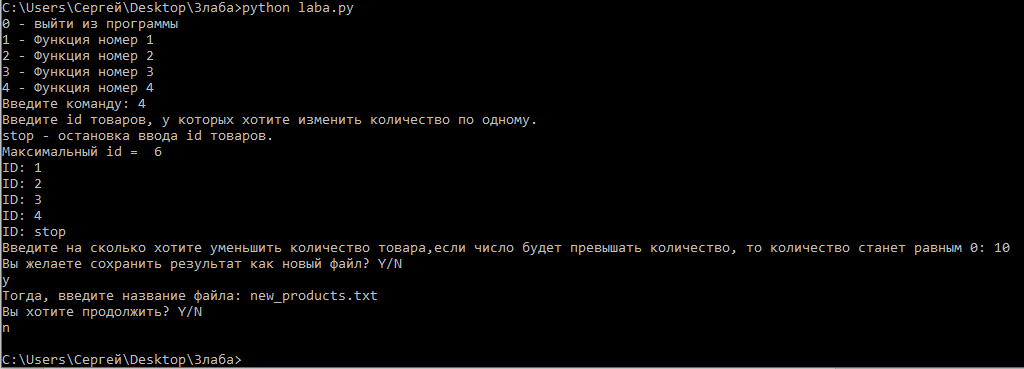


Рисунок 5 - выполнение 4 подпрограммы

Содержимое нового файла new\_products.txt представлено на рисунке 6

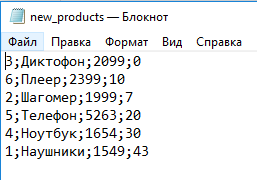


Рисунок 6 - содержимое нового файла new\_products.txt

* 1. Ссылка на репозиторий: https://github.com/IceUnite

1. Выполнив данную лабораторную работу, я получил опыт работы с чтением и записью в файл на ЯП Python.