Федеральное государственное бюджетное образовательное учреждение

высшего образования

«Уфимский государственный нефтяной технический университет»

Кафедра вычислительной техники и инженерной кибернетики

Задания по учебной практике №7-9

по дисциплине Информатика

«Программы обработки строковых данных.

Файлы и строки. Сортировка массивов данных»

Выполнил: студент гр. БПО09-24-02 Д.А. Хайруллин

Проверил: ст. преп. каф. ВТИК С.Г. Зайдуллина

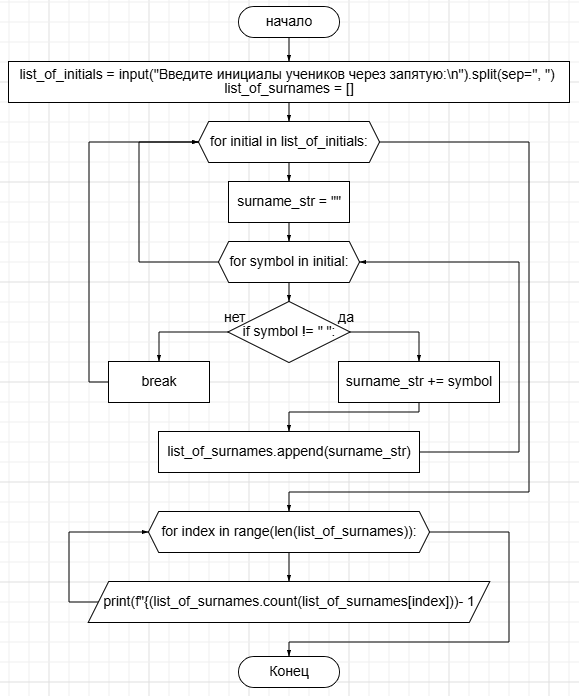
Уфа 2025

**Задача 1.Программы обработки строковых данных**

**Вариант 8**

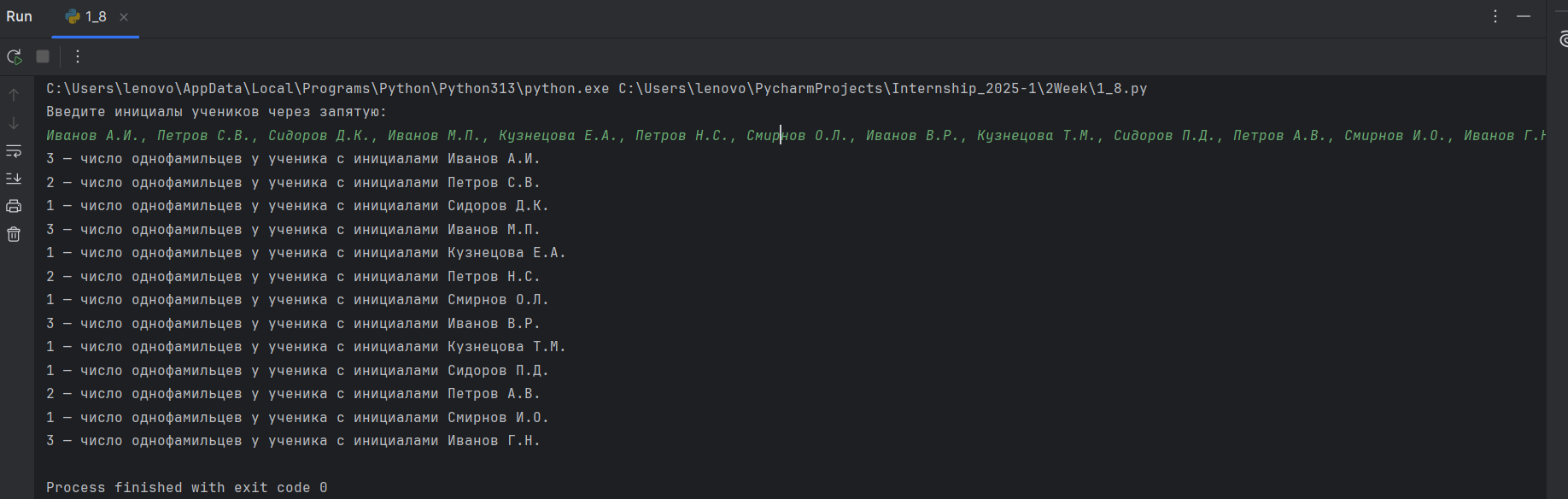
В массиве хранятся фамилии с инициалами учеников класса (Иванов П.П.). Требуется напечатать список класса с указанием для каждого ученика количества его однофамильцев.

**Блок-схема:**

****

**Код программы на Python:**

*# В массиве хранятся фамилии с инициалами учеников класса (Иванов П.П.).  
# Требуется напечатать список класса с указанием для каждого ученика количества его однофамильцев.*list\_of\_initials = input("Введите инициалы учеников через запятую:\n").split(sep=", ")  
list\_of\_surnames = []  
for initial in list\_of\_initials:  
 surname\_str = ""  
 for symbol in initial:  
 if symbol != " ":  
 surname\_str += symbol  
 else:  
 break  
 list\_of\_surnames.append(surname\_str)  
  
for index in range(len(list\_of\_surnames)):  
 print(f"{(list\_of\_surnames.count(list\_of\_surnames[index]))- 1} — число однофамильцев у ученика с инициалами {list\_of\_initials[index]}")  
  
**Результат:**

****

**Задача 2. Файлы и строки**

Создать программу, позволяющую из одного текстового файла, содержащего несколько строк (тип String) текста на русском языке, построчно переписать в другой текстовый файл слова, отвечающие некоторому условию. Задания выполнять согласно вариантам.

Требования:

– в новом файле следует указать номер строки, в которой искомые слова находились в исходном файле;

– для каждой строки в конце указать количество выбранных слов.

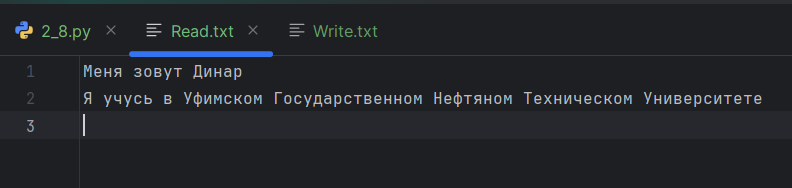
**Вариант 8**

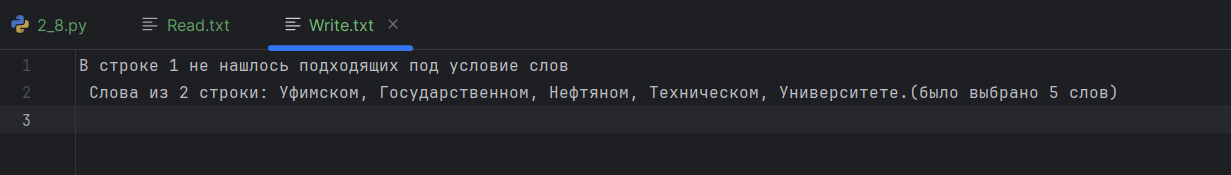
Переписать в результирующий файл слова с числом букв больше 5.

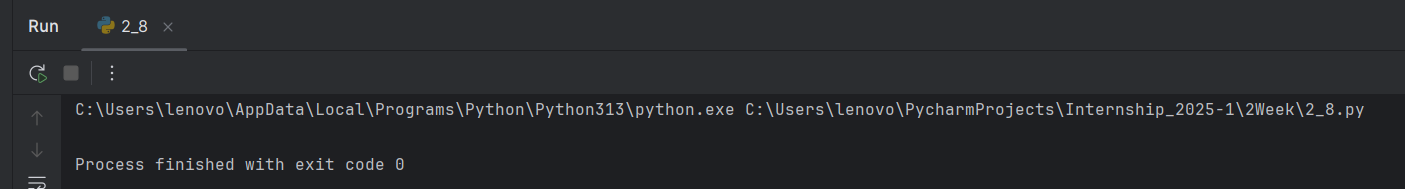
**Код программы на Python:**

*# Создать программу, позволяющую из одного текстового файла, содержащего несколько строк (тип String)  
# текста на русском языке, построчно переписать в другой текстовый файл слова, отвечающие некоторому условию.  
# Задания выполнять согласно вариантам.  
# Требования:  
# – в новом файле следует указать номер строки, в которой искомые слова находились в исходном файле;  
# – для каждой строки в конце указать количество выбранных слов.  
  
# Переписать в результирующий файл слова с числом букв больше 5.*with open ("Read.txt", "r", encoding="utf-8") as read\_the\_file:  
 with open("Write.txt", "w", encoding="utf-8") as write\_the\_file:  
 i = 1  
 while True:  
 line = read\_the\_file.readline()  
 new\_line = ""  
 if line == "":  
 break  
 new\_line += f"Слова из {i} строки: "  
 list\_of\_line = line.split()  
 number\_of\_words = 0  
 for word in list\_of\_line:  
 if len(word) > 5:  
 new\_line += word + ", "  
 number\_of\_words += 1  
  
 if len(new\_line) == 19:  
 new\_line = f"В строке {i} не нашлось подходящих под условие слов\n"  
 write\_the\_file.write(new\_line)  
 else:  
 new\_line = new\_line[:-2] + "."  
 write\_the\_file.write(f" {new\_line}(было выбрано {number\_of\_words} слов)\n")  
 i += 1

**Результат:**

****

****

****

**Задача 3**

**Вариант 8**

В магазине для упаковки подарков есть N кубических коробок разной стоимости. Самой интересной считается упаковка подарка по принципу матрёшки: подарок упаковывается в одну из коробок, та в свою очередь в другую коробку и т. д. Одну коробку можно поместить в другую, если длина её стороны хотя бы на K единиц меньше длины стороны другой коробки. Определите наибольшее количество коробок, которое можно использовать для упаковки одного подарка, так чтобы стоимость упаковки не превысила M единиц, и минимальную итоговую стоимость этой упаковки Размер подарка позволяет поместить его в самую маленькую коробку. Если есть несколько вариантов упаковки с одинаковым наибольшим количеством коробок, выберите вариант с наименьшей стоимостью.  
**Входные данные** представлены в файле 8.txt следующим образом. В первой строке входного файла находится число N – количество коробок в магазине (натуральное число, не превышающее 10 000), число K – минимально допустимая разница длин сторон соседних коробок в матрёшке, и число M – максимально допустимая стоимость упаковки. В каждой из следующих N строк записаны длина стороны коробки (натуральное число, не превышающее 10 000) и стоимость коробки.  
Запишите в ответе два целых числа: наибольшее количество коробок, которое можно использовать для упаковки одного подарка, так чтобы стоимость упаковки не превысила M единиц, и минимальную итоговую стоимость упаковки.  
**Пример входного файла**:

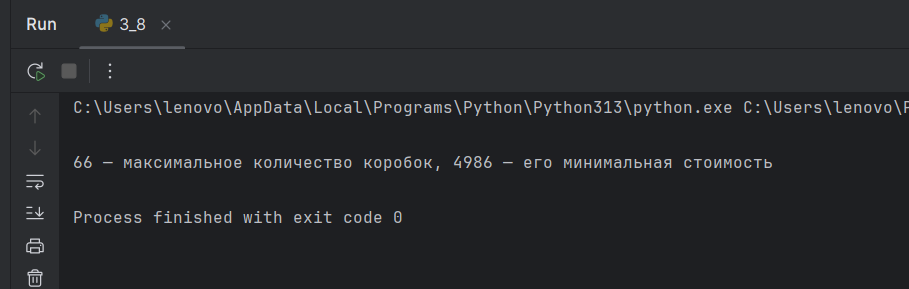
5 3 10  
50 5  
40 6  
30 5  
20 3  
10 15

При таких исходных данных максимальное количество коробок (2) при минимальной стоимости (8) получается при использовании коробок со сторонами 50 и 20. Ответ: 2 8.

**Код программы на Python**

with open('8.txt', 'r', encoding="utf-8") as file:  
 N, K, M = map(int, file.readline().split())  
 print()  
 boxes = []  
 for \_ in range(N):  
 size, cost = map(int, file.readline().split())  
 boxes.append((size, cost))  
boxes.sort(reverse=True, key=lambda x: x[0])  
max\_count = 0  
min\_cost = 1000000000000000000  
for i in range(N):  
 current\_size, current\_cost = boxes[i]  
 if current\_cost > M:  
 continue  
 count = 1  
 total\_cost = current\_cost  
 last\_size = current\_size  
 for j in range(i + 1, N):  
 next\_size, next\_cost = boxes[j]  
 if last\_size - next\_size >= K and total\_cost + next\_cost <= M:  
 count += 1  
 total\_cost += next\_cost  
 last\_size = next\_size  
 if count > max\_count or (count == max\_count and total\_cost < min\_cost):  
 max\_count = count  
 min\_cost = total\_cost  
print(f"{max\_count} — максимальное количество коробок, {min\_cost} — его минимальная стоимость")

**Результат:**

****

**Выводы по работе.**

В ходе выполнения лабораторной работы я закрепил знания о обработке строк, работе с файлами и строками, обработке массивов данных в рамках языка Python.