**Задачи на сортировки, списки и словари**

1. Дан список, состоящий из строк. Отсортировать его по длине слов. Сначала должны идти длинные слова затем короткие.
2. Дан список состоящий из слов. Отсортировать его по количеству вхождений буквы ***'a'***
3. Создайте словарь, связав его с переменной school, и наполните данными, которые бы отражали количество учащихся в разных 9 классах (9а, 9б, 9в, 9м, 9ф и т. п.). Внесите изменения в словарь согласно следующему:

а) в одном из классов изменилось количество учащихся

б) в школе появился новый класс.

в) в школе был расформирован (удален) другой класс.

г) Вычислите общее количество учащихся 9 классов в школе.

1. Коля устал запоминать телефонные номера и заказал у Вас программу, которая заменила бы ему телефонную книгу. Коля может послать программе два вида запросов: строку, содержащую имя контакта и его номер, разделенные пробелом, или просто имя контакта. В первом случае программа должна добавить в книгу новый номер, во втором – вывести номер контакта. Ввод происходит до символа точки. Если введенное имя уже содержится в списке контактов, необходимо перезаписать номер.

**Sample Input:**

**Ben 89001234050**

**Mary**

**Alice 210-220**

**Alice**

**Alice 404-502**

**Ben**

**Nick +4(908)273-22-42**

**Nick**

**Alice**

**Robert 51234047129**

**.**

**Sample Output:**

**не найдено**

**210-220**

**89001234050**

**+4(908)273-22-42**

**404-502**

**Задачи на исключения и декораторы**

1. Написать функцию **get\_element(lst: list, index:int),** которая возвращает элемент списка lst по индексу index. Если индекс выходит за границы списка, вернуть IndexError "Ошибка: индекс вне диапазона".
2. Написать декоратор **retry\_on\_exception(retries),** который повторяет вызов функции retries раз, если выбрасывается исключение ValueError.
3. Создать декоратор **timeout(limit),** который выбрасывает исключение TimeoutError, если функция выполняется дольше limit секунд.

**Дополнительные задачи:**

1. Когда Антон прочитал «Войну и мир», ему стало интересно, сколько слов и в каком количестве используется в этой книге. Помогите Антону написать упрощённую версию такой программы, которая сможет подсчитать слова, разделённые пробелом и вывести получившуюся статистику. Программа должна считывать одну строку со стандартного ввода и выводить для каждого уникального слова в этой строке число его повторений (без учёта регистра) в формате "слово количество". Порядок вывода слов может быть произвольным, каждое уникальное слово﻿ должно выводиться только один раз.

**Sample Input 1:**

**a aa abC aa ac abc bcd a**

**Sample Output 1:**

**ac 1**

**a 2**

**abc 2**

**bcd 1**

**aa 2**

**Sample Input 2:**

**a A a**

**Sample Output 2:**

**a 3**

1. Создать декоратор **cache\_results**, который запоминает результаты функции. Если функция вызывается с теми же аргументами, она не выполняется снова, а возвращается закешированный результат.