

Лабораторная работа №2

Методы работы с коллекциями и строками. Введение в анализ данных

Задача 1 (2 балла). Дистанционный датчик посылает сообщения, состоящие из заглавных латинских букв (от А до Z). Например, сообщение может выглядеть так: 'AAACBVBVBDDADDD'. Такое сообщение легко сжать, если заменить группу одинаковых символов на число, равное количеству повторов символа, и сам этот символ. То есть, например, группа 'AAA' заменится на '3A'. При этом одиночные буквы заменять не нужно, чтобы не произошло увеличения длины. В результате такого преобразования строка, приведенная в примере, преобразуется в '3AC5B2DA3D'. Напишите программу, выполняющую указанное преобразование.

Входные данные

В единственной строке записано исходное сообщение. Сообщение содержит только большие латинские символы. Длина сообщения – не менее 1 и не более 200 символов.

Выходные данные

В единственной строке выведите сжатое сообщение.

Пример

Входные данные	Выходные данные
AAACBVBVBDDADDD	3AC5B2DA3D

Задача 2 (2 балла). Пароль считается надежным, если его длина составляет не менее 12 символов, при этом он должен содержать хотя бы одну заглавную букву, хотя бы одну строчную букву, хотя бы одну цифру и хотя бы один спецсимвол. Любые другие символы в пароле запрещены. Напишите программу, которая по указанному списку паролей определяет, какие из них являются надежными, а какие – нет.

Допустимые заглавные буквы: 'A', 'B', ..., 'Z' (все символы латинского алфавита).

Допустимые строчные буквы: 'a', 'b', ..., 'z' (все символы латинского алфавита).

Допустимые спецсимволы: '!', '@', '#', '\$', '%', '&', '*', '+', ...

Входные данные

В первой строке записано одно целое число N – количество паролей, которые нужно проверить, $1 \leq N \leq 100$.

Далее следует N строк, в каждой из которых записан один пароль. Гарантируется, что в паролях нет недопустимых символов, а длина каждого – не больше 100 символов.

Выходные данные

Выведите N строк, по одной для каждого пароля из входных данных. В каждой выведите "Valid" (без кавычек), если пароль надежный, и "Invalid" (без кавычек), если ненадежный.

Пример

Входные данные	Выходные данные
2	Valid
123456789aA!	Invalid
123IsAVeryDifficultPassword	

Задача 3 (3 балла). Руководство Китежградского университета Чародейства и Волшебства считает, что аудиторный фонд используется нерационально – для проведения занятий выделены большие аудитории, но многие студенты прогуливают занятия, в результате в аудиториях остается много свободных мест. Чтобы убедиться в этом, было решено по каждой группе на каждой паре отмечать количество присутствующих студентов. Тут выяснилась еще одна проблема – некоторые преподаватели недобросовестно отнеслись к этому поручению, и отмечали иногда явно нереальные значения – то 0 студентов, то в два раза больше, чем вообще есть в группе.

Ваша задача – по имеющимся записям о посещаемости одной группы определить, какая доля всех записей является некорректной.

Некорректными мы будем считать данные, которые сильно отличаются от остальных (в статистике это называется *выбросами*). Более точно алгоритм выглядит так. Пусть у нас есть набор данных, состоящий из N значений. Вначале найдем квартили для этого набора (Q_1 , Q_2 , Q_3 , см. справку ниже). Затем определим разброс квартилей $IQR = Q_3 - Q_1$. Выбросами (некорректными данными) будем считать все значения, которые меньше, чем $Q_1 - 1.5 \cdot IQR$, а также все значения, которые больше, чем $Q_3 + 1.5 \cdot IQR$.

Справка

Квартили – это значения, которые разбивают упорядоченный набор чисел на 4 равных (по количеству чисел) части. Для их нахождения в случае дискретного набора данных используют следующий алгоритм. Вначале найдем вторую квартиль Q_2 (она же – медиана). Отсортируем все числа. Пусть в наборе N значений: x_1, x_2, \dots, x_N . Если N нечетное, то Q_2 равно значению, стоящему ровно в середине (то есть $Q_2 = x_{(N+1)/2}$). Если же N четное, то Q_2 равно среднему арифметическому двух чисел, стоящих в середине (то есть $Q_2 = (x_{N/2} + x_{N/2+1})/2$). После этого разделим упорядоченный набор на две равные половины (если N нечетное, то средний элемент, равный Q_2 , не включается ни в одну из половин). Числа, которые меньше Q_2 , попадут в «нижнюю» половину, а числа, которые больше Q_2 – в «верхнюю». Тогда Q_1 – это медиана «нижней» половины, а Q_3 – медиана «верхней» половины.

Входные данные

В первой строке записано число N – количество пар, на которых отмечалась посещаемость группы, $5 \leq N \leq 1000$.

Далее следует N строк, в каждой записано одно целое число x_i – количество студентов, присутствовавших на i -й паре по мнению преподавателя, $0 \leq x_i \leq 100$.

Выходные данные

Выведите одно целое число – сколько чисел из представленных можно считать некорректными.

Пример

Входные данные	Выходные данные
9 25 24 19 99 18 17 0 21 26	2

Комментарий к примеру

После сортировки получим последовательность чисел: 0, 17, 18, 19, 21, 24, 25, 26, 99. По алгоритму находим $Q_2 = 21$. «Нижняя» половина – это числа 0, 17, 18, 19. Соответственно, $Q_1 = (17+18)/2 = 17.5$. «Верхняя» половина – это числа 24, 25, 26, 99. $Q_3 = (25+26)/2 = 25.5$. Тогда $IQR = Q_3 - Q_1 = 25.5 - 17.5 = 8$. Таким образом, некорректными в данной задаче будут считаться значения, которые меньше, чем $17.5 - 1.5 \cdot 8 = 17.5 - 12 = 5$ (такое значение всего одно – число 0), а также которые больше, чем $25.5 + 1.5 \cdot 8 = 25.5 + 12 = 37.5$ (такое значение тоже всего одно – число 99). Итого, всего 2 числа (0 и 99) считаются некорректными.

Задача 4 (3 балла). Компания друзей собралась пойти в поход. Забот и затрат при подготовке похода оказалось много: кто-то закупал еду, кто-то брал в аренду снаряжение, кто-то заказывал транспорт. Когда всё было готово, друзья решили подсчитать, кто сколько денег потратил и, соответственно, кто кому сколько денег должен перевести. Статей расходов оказалось очень много, участников похода было тоже много, поэтому сделать все расчеты вручную оказалось затруднительно.

Напишите программу, которая по информации о том, кто сколько денег потратил, определит: кто, кому и сколько денег должен перевести, чтобы расходы всех участников похода оказались одинаковыми (с точностью до копейки). Количество переводов при этом должно быть как можно меньше.

Входные данные

В первой строке через пробел записаны имена всех участников похода. Имена уникальны, каждое имя состоит из латинских букв. Длина каждого имени – не более 20 символов, количество имен – не более 100.

Во второй строке записано одно целое число N – количество покупок, которое было сделано при подготовке похода.

Далее следует N строк, каждая из которых описывает одну покупку и содержит имя того, кто эту покупку оплачивал, и одно целое число – сумму покупки. Имя и число разделены пробелом. Гарантируется, что имя есть в общем списке участников похода.

Выходные данные

В первой строке выведите одно число M – минимальное количество переводов, которые нужно совершить.

Далее выведите M строк, в каждой указав два имени и вещественное число через пробел: кто, кому и сколько должен перевести. Все суммы переводов должны быть округлены до 2 знаков после запятой. Если существует несколько вариантов переводов, то выведите любой из них – главное, чтобы их количество было минимальным.

Пример

Входные данные	Выходные данные
Ivan Aleksej Igor 3 Ivan 500 Aleksej 100 Ivan 200	2 Aleksej Ivan 166.67 Igor Ivan 266.67

Комментарий к примеру

Иван потратил в сумме 700, но двое остальных переведут ему в сумме 433.34, итого затраты Ивана составят 266.66. Алексей потратил 100 и должен перевести Ивану 166.67, в итоге его затраты составят 266.67. Игорь ничего не потратил на покупки, и все его затраты – это перевод 266.67 Ивану. Таким образом, затраты всех трех друзей окажутся одинаковыми с точностью до копейки (затраты Ивана на 1 копейку меньше, чем у остальных). Очевидно, что одним переводом добиться равных затрат не получится, поэтому 2 – это минимальное количество переводов.

Заметим, что такой набор переводов – не единственно возможный. Например, Алексей мог бы перевести 166.67 Игорю, а Игорь перевести Ивану 433.34

Задача 5 (3 балла). Начинающий предприниматель Александр открыл свою первую пиццерию. Для учета заказов он использует максимально простой инструмент – записывает в блокнот информацию о дате заказа, названии пиццы и стоимости заказа (стоимость одной и той же пиццы даже в один и тот же день может быть разной – это зависит от дополнительных ингредиентов, которые пожелал добавить клиент, но которые Александр в своем блокноте никак не учитывает). По прошествии нескольких дней Александр хочет извлечь из своих записей какую-нибудь полезную информацию. Напишите программу, которая будет выводить:

а) список всех пицц с указанием, сколько раз их заказывали; список должен быть отсортирован по убыванию количества заказов, то есть первой в списке должна оказаться самая популярная пицца;

б) список всех дат с указанием суммарной стоимости проданных в этот день пицц; список должен быть отсортирован хронологически;

в) информацию о самом дорогом заказе;

г) среднюю стоимость заказа (среднее арифметическое по всем стоимостям).

Формат входных и выходных данных определите самостоятельно.

Задача 6 (3 балла). Файл mbox.txt содержит метаданные почтового сервера. Мы знаем, что строка с адресом автора письма начинается с "From ". Найти адреса всех авторов сообщений и найти того из них, кто пишет больше всех писем.

Исходный файл можно взять по ссылке: <http://www.py4inf.com/code/mbox.txt>

Для создания локальной программы вы можете загрузить файл себе на компьютер. Либо можно выполнять это задание в Jupyter Notebook. Для загрузки данных можно использовать следующий код:

```
# импортируем библиотеку для доступа к файлам в интернете
import requests
# в переменной mbox хранится текст для работы
mbox = requests.get('http://www.py4inf.com/code/mbox.txt').text

# преобразуем текст в список, где каждый объект – это одна строка в файле
all_lines = mbox.split('\n')
```