

Лабораторная работа №1

Методы работы с коллекциями и строками

Задача 1. 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>

Задача 2. Предложите свою хеш-функцию для слов на естественном языке и реализуйте словарь с использованием этой функции. На входе – последовательность команд двух видов:

- `add <слово>` - добавить `<слово>` в словарь;
- `check <слово>` - проверить, что `<слово>` есть в словаре.

Для каждой команды вида `check` нужно выдать ответ: `yes` или `no`.

Задача 3. Начинаящий 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).

Поскольку исходный текстовый файл заполняется вручную, то формат данных может быть разным (даже в рамках одного файла). Гарантируется только, что каждая строка файла содержит информацию об одном заказе, и в каждой строке указаны 3 значения: дата, название и стоимость. Нужно учесть следующие варианты:

- значения могут быть разделены запятой, точкой с запятой или просто пробелом;
- в дате дата, месяц и год могут быть разделены точкой или косой чертой (/);
- название пиццы может быть указано в кавычках или без них;
- стоимость может быть целым или дробным числом (если число дробное, то в нем не более 2 знаков в дробной части, при этом дробная часть от целой может быть отделена точкой или запятой).

Если строка не соответствует указанному формату (с учетом указанных вариантов), то такую строку нужно пропускать.

Задача 4*. 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 (если сумма не делится нацело на количество участников, то затраты могут отличаться на 1 копейку). 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