D. Join между файлами

|  |  |
| --- | --- |
| Ограничение времени | 6 секунд |
| Ограничение памяти | 128.0 Мб |
| Ввод | стандартный ввод или input.txt |
| Вывод | стандартный вывод или output.txt |

Во время проектирования распределенной системы планировалось использовать две различные базы данных. Разработчики не предусмотрели, что им может понадобится выполнять операцию сопоставления значений между таблицами этих баз данных. Помогите им придумать, как выполнить операцию **[inner join](https://ru.wikipedia.org/wiki/Join_(SQL)" \l "INNER_JOIN)**.

База данных **market** и таблица **market.shop** представляет собой таблицу магазинов из двух колонок

* shop\_id - идентификатор магазина
* shop\_name - название магазина

База данных **billing** и таблица **billing.order** представляет собой таблицу заказов из трех колонок

* order\_id - идентификатор заказа
* shop\_id - идентификатор магазина для, который выполнил заказ
* cost - общая стоимость товаров в заказе

Необходимо получить новую таблицу в которой будут следующие колонки:

* order\_id - номер заказа
* shop\_name - название магазина
* shop\_id - идентификатор магазина
* cost - общая стоимость товаров в заказе

Формат ввода

Для удобства базы данных представлены в виде файлов в формате **csv**, каждый файл отсортирован по возрастанию поля **shop\_id**.

Содержимое файлов, используемых в тестах условия, представлено **в примечании**.

На вход программе подаются названия файлов в одной строке, разделенные пробелом, в которых содержатся таблицы, описанные ниже. Строка заканчивается переводом строки (символом '\n').

market.shop.filename billing.order.filename

Пример содержимого файла таблицы market.shop:

shop\_id,shop\_name

1,MyShop

2,Pleer.ru

...

Гарантируется, что все **shop\_id** в таблице market.shop **уникальны**.

Пример содержимого файла таблицы billing.order:

order\_id,shop\_id,cost

16423,1,23040

242,2,1680000

...

Гарантируется, что все **order\_id** в таблице billing.order **уникальны**.

Обозначим за N*N*, M*M* количество строк с данными в файлах market.shop.filename и billing.order.filename соответственно.

Гарантируются следующие ограничения:

0≤N,M≤1060≤*N*,*M*≤106.

0≤0≤ shop\_id, order\_id, cost ≤1018≤1018

1≤1≤ |shop\_name| ≤30≤30, значения в колонке shop\_name состоят из латинских букв, цифр и знаков подчеркивания.

Формат вывода

Таблица в виде

order\_id,shop\_name,shop\_id,cost

16423,MyShop,1,23040

242,Pleer.ru,2,1680000

...

Пример 1

| **Ввод**  Скопировать ввод | **Вывод**  Скопировать вывод |
| --- | --- |
| market\_1.csv billing\_1.csv | order\_id,shop\_name,shop\_id,cost  1,shop\_name\_1,1,100  2,shop\_name\_2,2,200  3,shop\_name\_3,3,300  4,shop\_name\_4,4,400  5,shop\_name\_5,5,500 |

Пример 2

| **Ввод**  Скопировать ввод | **Вывод**  Скопировать вывод |
| --- | --- |
| market\_1.csv billing\_2.csv | order\_id,shop\_name,shop\_id,cost  1,shop\_name\_1,1,100  2,shop\_name\_2,2,200  3,shop\_name\_2,2,200  4,shop\_name\_2,2,200  5,shop\_name\_5,5,500 |

Пример 3

| **Ввод**  Скопировать ввод | **Вывод**  Скопировать вывод |
| --- | --- |
| market\_2.csv billing\_1.csv | order\_id,shop\_name,shop\_id,cost  2,shop\_name\_2,2,200 |

Пример 4

| **Ввод**  Скопировать ввод | **Вывод**  Скопировать вывод |
| --- | --- |
| market\_2.csv billing\_2.csv | order\_id,shop\_name,shop\_id,cost  2,shop\_name\_2,2,200  3,shop\_name\_2,2,200  4,shop\_name\_2,2,200 |

Примечания

**market\_1.csv**

shop\_id,shop\_name

1,shop\_name\_1

2,shop\_name\_2

3,shop\_name\_3

4,shop\_name\_4

5,shop\_name\_5

**market\_2.csv**

shop\_id,shop\_name

2,shop\_name\_2

**billing\_1.csv**

order\_id,shop\_id,cost

1,1,100

2,2,200

3,3,300

4,4,400

5,5,500

**billing\_2.csv**

order\_id,shop\_id,cost

1,1,100

2,2,200

3,2,200

4,2,200

5,5,500