

# Диагностика технических навыков (backend)

11 авг 2023, 16:10:30

старт: 10 авг 2023, 20:25:58

финиш: 10 авг 2023, 23:25:58

длительность: 03:00:00

начало: 1 дек 2021, 02:00:00

## С. Опять JSON'ы перекладывать...

Ограничение времени	2 секунды
Ограничение памяти	256.0 Мб
Ввод	стандартный ввод или input.txt
Вывод	стандартный вывод или output.txt

Всем хотя бы раз в жизни приходилось перекладывать JSON. Вот и для нового проекта под названием "Единое хранилище" необходимо переложить магазинные фиды. Для размещения на Яндекс.Маркете магазины передают товары из своего ассортимента в JSON-файлах. Одно товарное предложение описывается так:

```
{
  "offer_id": <string>,
  "market_sku": <int>,
  "price": <int>
}
```

где *offer\_id* - уникальный среди всех фидов магазина идентификатор предложения, *market\_sku* - идентификатор товара на Яндекс.Маркете, *price* - стоимость товара.

Весь фид магазина представляет собой JSON и выглядит так:

```
{
  "offers": [<offer>, <offer>, ...]
}
```

Вас попросили написать программу, которая объединит все фиды одного магазина в единый фид и отсортирует товары в порядке неубывания их стоимостей, а при их равенстве - по *offer\_id*.

### Формат ввода

В первой строке входных данных содержится целое число  $n$  - количество фидов магазина ( $1 \leq n \leq 200$ ). Следующие  $n$  строк содержат по одному магазинному фиду на строку. Гарантируется, что строка - валидный JSON и удовлетворяет формату фида. В одном фиде не больше 200 товарных предложений. *offer\_id* состоит из строчных и заглавных букв латинского алфавита и цифр,  $1 \leq |offer\_id| \leq 10$ ,  $1 \leq market\_sku \leq 2^{31} - 1$ ,  $1 \leq price \leq 10^6$ .

### Формат вывода

Выходной поток должен содержать один JSON-документ, удовлетворяющий формату товарного фида. Количество строк в выходном файле и табуляция не имеют значения.

## Пример

### Ввод

```
2
{"offers": [{"offer_id": "offer1", "market_sku": 10846332, "price": 1490}, {"offer_id": "offer2", "market_sku": 682644, "pr
{"offers": [{"offer_id": "offer3", "market_sku": 832784, "price": 14000}]}
```

### Вывод

```
{"offers":[{"market_sku":682644,"offer_id":"offer2","price":499}, {"market_sku":10846332,"offer_id":"offer1","price":1490}, {
```

## Примечания

Для решений на языке **python** доступны все стандартные библиотеки, включая json.

Для решений на языке **Java** доступна библиотека [json-simple](#) версии 1.1.1. Соответствующие import'ы могут выглядеть так:

```
import org.json.simple.JSONArray;
import org.json.simple.JSONObject;
import org.json.simple.parser.JSONParser;
import org.json.simple.parser.ParseException;
```

Для решений на языке **C++** доступна библиотека [nlohmann/json](#) v3.8.0. Соответствующие include выглядят так:

```
#include "json.hpp"
```

Для решений на **golang** доступны все стандартные пакеты, включая encoding/json, sort и другие.

Язык

Python 3.7 + network + r...

Набрать здесь

Отправить файл

Выбрать

Файл не выбран

Отправить

Предыдущая

Следующая