



## СОРЕВНОВАНИЯ

[ЗАДАЧИ](#) [ОТОСЛАТЬ](#) [МОИ ПОСЫЛКИ](#) [СТАТУС](#) [ПОЛОЖЕНИЕ](#) [ЗАПУСК](#)

## F. JSON категории (25 баллов)

ограничение по времени на тест: 1 секунда

ограничение по памяти на тест: 256 мегабайт

ввод: стандартный ввод

вывод: стандартный вывод

На маркетплейсе у каждого товара есть категория. При этом, у некоторых категорий есть дочерние категории. Для удобной навигации по маркетплейсу покупатели могут пользоваться деревом категорий.

## ВСЕ ТОВАРЫ

&gt; Одежда

&gt; Электроника

&gt; Компьютеры

Моноблоки

&gt; Телевизоры

&gt; Дом и ремонт

Ваша задача — построить дерево категорий. Дана информация об отношениях родительских и дочерних категорий в виде JSON-массива. Каждый элемент массива является словарем, с полями `name` (название категории), `id` (числовой идентификатор категории) и `parent` (числовой идентификатор родительской категории). Известно, что корневая категория имеет нулевой идентификатор и не имеет идентификатора родительской категории.

По данной информации постройте дерево категорий в виде JSON-словаря. Словарь для каждой категории должен иметь поля `name`, `id` и массив `next`, состоящий из таких же словарей для дочерних категорий.

## Входные данные

Каждый тест состоит из нескольких наборов входных данных. Первая строка содержит целое число  $t$  ( $1 \leq t \leq 100$ ) — количество наборов входных данных. Далее следует описание наборов входных данных.

Первая строка каждого набора входных данных содержит целое число  $n$  ( $1 \leq n \leq 1000$ ) — количество строк с описанием JSON-массива категорий.

Следующие  $n$  строк содержат описание JSON-массива категорий. Все числовые идентификаторы категорий являются целыми числами и удовлетворяют условию  $0 \leq id \leq 10^9$ . Все имена категорий непустые, состоят из строчных латинских букв и имеют длину не больше 20. В описании могут быть символы пробела и табуляции.

Гарантируется, что каждый набор входных данных содержит корневую категорию и не более 400 категорий.

Гарантируется, что размер входных данных не превосходит 10Мб.

## Выходные данные

Выведите JSON-массив из  $t$  элементов.  $i$ -й элемент массива является словарем с описанием дерева категорий для  $i$ -го набора входных данных. При проверке ответа пробелы, табы и переносы строки не учитываются (кроме таковых в json полях). Порядок полей в словаре и порядок дочерних категорий в массиве `next` не учитывается. Если у

## Route 256: Junior

Участник

[→ О группе](#)[→ Соревнования группы](#)

- Дорешивание
- Контекст - C# (Juniors)
- Контекст - Go (Juniors)
- Песочница (C#)
- Песочница (Go)

## Контекст - Go (Juniors)

Закончено

Участник

[→ Пересчёт ограничений по времени](#)

Это соревнование использует политику пересчёта ограничений по времени по языкам программирования. Система автоматически увеличивает ограничения по времени для некоторых языков в соответствии с множителями. Независимо от множителя языка, ограничение по времени не может превысить 30 секунд. Прочтите детали [по ссылке](#).

[→ Языки](#)

Только перечисленные языки могут быть использованы для решения задач соревнования

## Контекст - Go (Juniors):

- GNU GCC C11 5.1.0
- Clang++20 Diagnostics
- Clang++17 Diagnostics
- GNU G++14 6.4.0
- GNU G++17 7.3.0
- GNU G++20 11.2.0 (64 bit, winlibs)

категории нет дочерних категорий, ключ next может отсутствовать, или соответствовать пустому массиву.

Примеры

входные данные

Скопировать

```
2
21
[
  {
    "id":0,
    "name":"all"
  },
  {
    "id":1,
    "name":"clothes",
    "parent":0
  },
  {
    "id":2,
    "name":"shoes",
    "parent":0
  },
  {
    "id":55,
    "name":"sneakers",
    "parent":2
  }
]
6
[ {"parent": 0,"id":100, "name":
"x"},{
"name":"x","id":0}
```

выходные данные

Скопировать

```
[{
  "id": 0,
  "name": "all",
  "next": [{
    "id": 1,
    "name": "clothes",
    "next": []
  }, {
    "id": 2,
    "name": "shoes",
    "next": [{
      "id": 55,
      "name": "sneakers"
    }]
  }]
},
{"name":"x","id":0,"next":[{"id":100,"name":"x"}]}
```

входные данные

Скопировать

```
1
9
[
{"name": "everything", "id": 0},
{"name": "clothes", "id": 1, "parent": 0},
{"name": "electronics", "id": 2, "parent": 0},
{"name": "computers", "id": 4, "parent": 2},
{"name": "aio", "id": 3, "parent": 4},
{"name": "tv", "id": 5, "parent": 2},
{"name": "house", "id": 6, "parent": 0}
]
```

выходные данные

Скопировать

```
[
{"id":0,"name":"everything","next":[{"id":1,"name":"clothes","next":[]},
{"id":2,"name":"electronics","next":[{"id":4,"name":"computers","next":
[{"id":3,"name":"aio","next":[]}]}, {"id":5,"name":"tv","next":[]}]},
{"id":6,"name":"house","next":[]}]}
```

Примечание

- Microsoft Visual C++ 2017
- GNU G++17 9.2.0 (64 bit, msys 2)
- C# 8, .NET Core 3.1
- C# 10, .NET SDK 6.0
- C# Mono 6.8
- Go 1.19.5
- Java 11.0.6
- Java 17 64bit
- Java 1.8.0\_241
- Delphi 7
- Free Pascal 3.0.2
- PascalABC.NET 3.8.3
- PHP 8.1.7
- PostgreSQL 15.1
- Python 2.7.18
- Python 3.8.10
- PyPy 2.7.13 (7.3.0)
- PyPy 3.6.9 (7.3.0)
- PyPy 3.9.10 (7.3.9, 64bit)

→ Последние посылки

Посылка	Время	Вердикт
<a href="#">220724005</a>	27.08.2023 16:49	Полное решение: 25 баллов
<a href="#">220722636</a>	27.08.2023 16:42	Частичное решение: 0 баллов
<a href="#">220720695</a>	27.08.2023 16:31	Частичное решение: 0 баллов
<a href="#">220719748</a>	27.08.2023 16:27	Частичное решение: 0 баллов
<a href="#">220718474</a>	27.08.2023 16:21	Частичное решение: 0 баллов
<a href="#">220714524</a>	27.08.2023 16:02	Частичное решение: 0 баллов
<a href="#">220710102</a>	27.08.2023 15:40	Частичное решение: 0 баллов
<a href="#">220709937</a>	27.08.2023 15:39	Частичное решение: 0 баллов

→ Набранные баллы

	Баллы
A	5
B	10
C	10
D	15
E	10
F	25
G	
H	
S1	
S2	
Всего	75

- Материалы соревнования
- problem-b-tests.zip
  - problem-c-tests.zip
  - problem-d-tests.zip
  - problem-e-tests.zip
  - problem-f-tests.zip
  - problem-g-tests.zip
  - problem-h-tests.zip
  - problem-s1-tests.zip
  - problem-s2-tests.zip

Любые внешние библиотеки использовать нельзя.

- Для работы с JSON в языке C# можно пользоваться библиотекой [System.Text.Json](#).  
Возможно, вам понадобится увеличить максимальную глубину сериализации/десериализации JSON ([документация](#)).
- Для работы с JSON в языке Go можно пользоваться библиотекой [encoding/json](#).
- Для работы с JSON в языке Python можно пользоваться библиотекой [json](#).

---

Codeforces (c) Copyright 2010-2023 Михаил Мирзаянов  
Соревнования по программированию 2.0  
Время на сервере: 11.09.2023 23:19:22 (11).  
Десктопная версия, переключиться на [мобильную](#).  
[Privacy Policy](#)

На платформе

