**Настроить автопилот**

Отправившись на встречу с бабушкой, чтобы попробовать её новые капустно-луковые пирожки, вы сели в первый попавшийся поезд: все пути, как известно, ведут куда надо.

Проезжая очередной неизвестный населенный пункт, вы задумываетесь: возможно, дороги ведут туда, куда ведут, а маршрут ещё нужно построить. Работники на ближайшей станции говорят вам, что сами не уверены, как эти маршруты строить, и просят вас помочь - они предоставят вам карту доступных маршрутов, но им нужна программа автопилот.

"Да это же точно такой принцип как у хлебных крошек на сайтах!" - восклицаете вы, припоминая свой опыт веб-разработчика.

**Формат ввода**

Ввод состоит из 3 объектов, которые будут переданы в функцию решения.

**routeTree**

Содержит древовидную структуру, описывающую возможные маршруты. Каждый элемент RouteNode имеет вид:

{

route: string, // сегмент маршрута

title: string, // имя маршрута

redirectTo: string, // сегмент маршрута, на который должно происходить перенаправление

children: RouteNode[] // массив дочерних элементов (при их наличии)

}

Таким образом **routeTree** представляет собой корневой элемент RouteNode с фиксированным значением route: ’/’.

Значение сегмента маршрута может иметь вид :someEntityId. В этом случае адрес маршрута содержит параметр на месте этого сегмента, который будет использоваться для получения имени маршрута с помощью набора данных с ключом someEntity. Поле title для таких сегментов не определено.

Поле redirectTo опциональное. Если оно задано, то должно содержать значение сегмента того же уровня, что и текущий. Ссылка и имя маршрута будут соответствовать сегменту перенаправления. Элемент перенаправляемого маршрута не содержит поля title, а элемент, на который указывает redirectTo не содержит дочерних элементов. Параметризованные сегменты не могут быть перенаправлены и не могут быть целью перенаправления.

**data**

Объект, ключи которого определяют имена наборов данных, а значения - наборы данных. Набор данных в свою очередь представлен объектом, ключи которого соответствуют идентификаторам сущностей, а значения - именам сущностей.

**urls**

Массив строк, где каждая строка содержит адрес маршрута (URL), по которому мы должны сгенерировать данные маршрута для автопилота.

**Формат вывода**

Массив, элементы которого являются ответами для соответствующего URL из входных данных.

Ответ - это массив объектов вида:

{

route: string, // адреc маршрута (URL)

title: string, // имя маршрута

}

**Примечание**

Файл с решением требуется оформить по шаблону:

module.exports = function (routeTree, data, urls) {

// ...

return result;

}

Не допускается использование минифицированного кода в решениях. Организаторы оставляют за собой право принимать решение о результатах соревнования на основе экспертной оценки исходного кода участников.

**Пример**

Пример входных и выходных данных можно скачать ниже [по ссылке](https://disk.yandex.ru/d/wC2sdvTTryLy4Q).