

# Web API

## А. Записная книжка

Записи в записной книжке распределены по различным сервисам в одной сети. В книжке хранятся имена и телефоны друзей в виде пары <name><cell phone number> (например, "Алексей 89215554433").

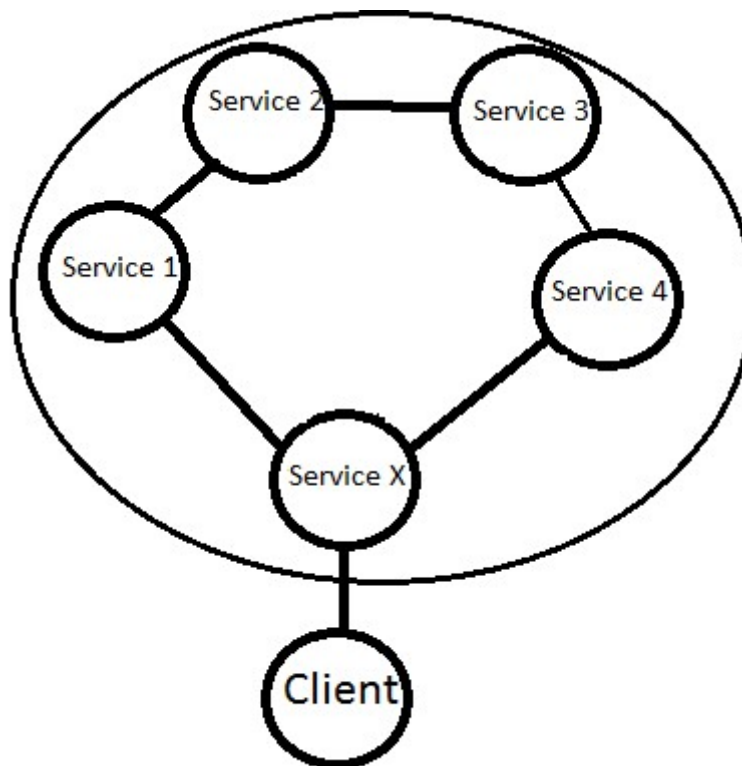
Вы не знаете заранее, какие данные на каком сервисе хранятся.

Необходимо найти запись в распределенной записной книжке по имени или телефону, которые вводит клиент. Если данные не найдены, сообщить об этом. Также клиент может добавлять новые записи в записную книжку и они должны распределяться равномерно по сервисам.

Все сервисы (узлы) связаны в кольцо по схеме Token Ring (см. рисунок). Всего пять сервисов.

Клиент находится вне этого кольца и обращается всегда к одному заранее известному узлу сети ServiceX (клиент знает адрес лишь этого сервиса).

Если сервис может обработать запрос (т.е. запись в его данных была найдена), он возвращает его тому сервису, от которого запрос пришел (и так далее по цепочке результат запроса возвращается к клиенту). Иначе передает тот же запрос другому сервису, с которым он соединен. Если запрос прошел по кругу и ничего не было найдено, то клиент получает соответствующее уведомление (это не Exception).



Другие требования:

1. Каждый сервис внутри кольца должен распознавать своего левого и правого соседа и быть доступен только для них. Т.е., например, ServiceX не сможет обратиться к Service 2, даже если знает его адрес и API. Сервисы должны узнавать друг друга по токенам доступа. Токены обновляются автоматически каждый раз, когда стартует ServiceX.
2. Клиент должен авторизоваться перед тем как начать пользоваться словарем

## Б. Циркулярная хеш-таблица

В качестве альтернативного задания вы можете реализовать свою циркулярную хеш-таблицу, которая будет работать на основе сервисов. Для примера пусть будут храниться данные записной книжки как в Задании 1.

### Общие требования (для заданий А и Б)

Реализовать сервисы необходимо на основе Web API

Продемонстрируйте работоспособность вашего решения на основе юнит-тестов (с использованием HttpClient)