



Строки

Лекция 5

План на сегодня

i

i

i

i

Z-функция

КМП через Z-функцию

Хеши и применения

Хеш-таблица

Z-функция

От строки это массив, где i -тый элемент равен длине максимальной подстроки, который начинается в этом элементе и совпадает с префиксом

a	b	a	c	a	b	a
7	0	1	0	3	0	1

Как насчитать такой массив за $O(n^2)$

Z-функция за $O(n)$

Будем использовать два указателя,
в которых поддерживаем
максимальный отрезок
совпадающий с префиксом

a	b	a	c	a	b	a
7	0	1	0	3	0	1

КМП через Z-функцию

Хотим найти все вхождения строки S в строку T .
Давайте сделаем новую строку $S+\#+T$, а потом насчитаем на ней Z-функцию

a b a # b a b a c

Как полученном массиве найти вхождения?

Хеш

Это какая-то функцию, которая детерминировано преобразует входные данные в выходные. К примеру, остаток от деления на 2 это тоже хеш

В криптографии хеши обязательно должны быть необратимыми т.е. криптографы должны придумать такую функцию по выводу которой нельзя получить ввод



Применения хешей

- Балансировка нагрузки
- Хранение паролей
- Проверка целостности файлов
- Криптография, блокчейны и так далее

Сейчас нас больше всего интересует как их использовать вместе со строками

Полиномиальный хеши

Допустим у нас есть строка S . Давайте возьмем число k , а все буквы представим как числа и будем умножать на k в степени индекса буквы.

Понятно, что числа могут получиться большими, поэтому возьмем результат по модулю

$$h_f = (s_0 + s_1k + s_2k^2 + \dots + s_nk^n) \bmod p$$

Как такой массив поможет нам сравнивать строки?

Коллизии

Из-за того, что мы берем все числа по модулю, то одинаковые строки иногда могут давать одинаковый хеш. Более того, зная пару p и k можно подобрать тест, на котором хеш не сработает. Будьте осторожны!

Как можно с этим бороться:

- Брать простые числа p и k – в таком случае вероятность коллизии меньше
- Считать несколько хешей по разным модулям

В общем случае:

Если ваш $p > n^2$, то скорее всего коллизий не будет

$$h_f = (s_0 + s_1k + s_2k^2 + \dots + s_nk^n) \bmod p$$

Хеши + бинпоиск

Это очень крутая комбинация, которая
позволяет сравнивать строки

aabbcabca
aabbssaca

Хеши + бинпоиск

Это очень крутая комбинация, которая
позволяет сравнивать строки

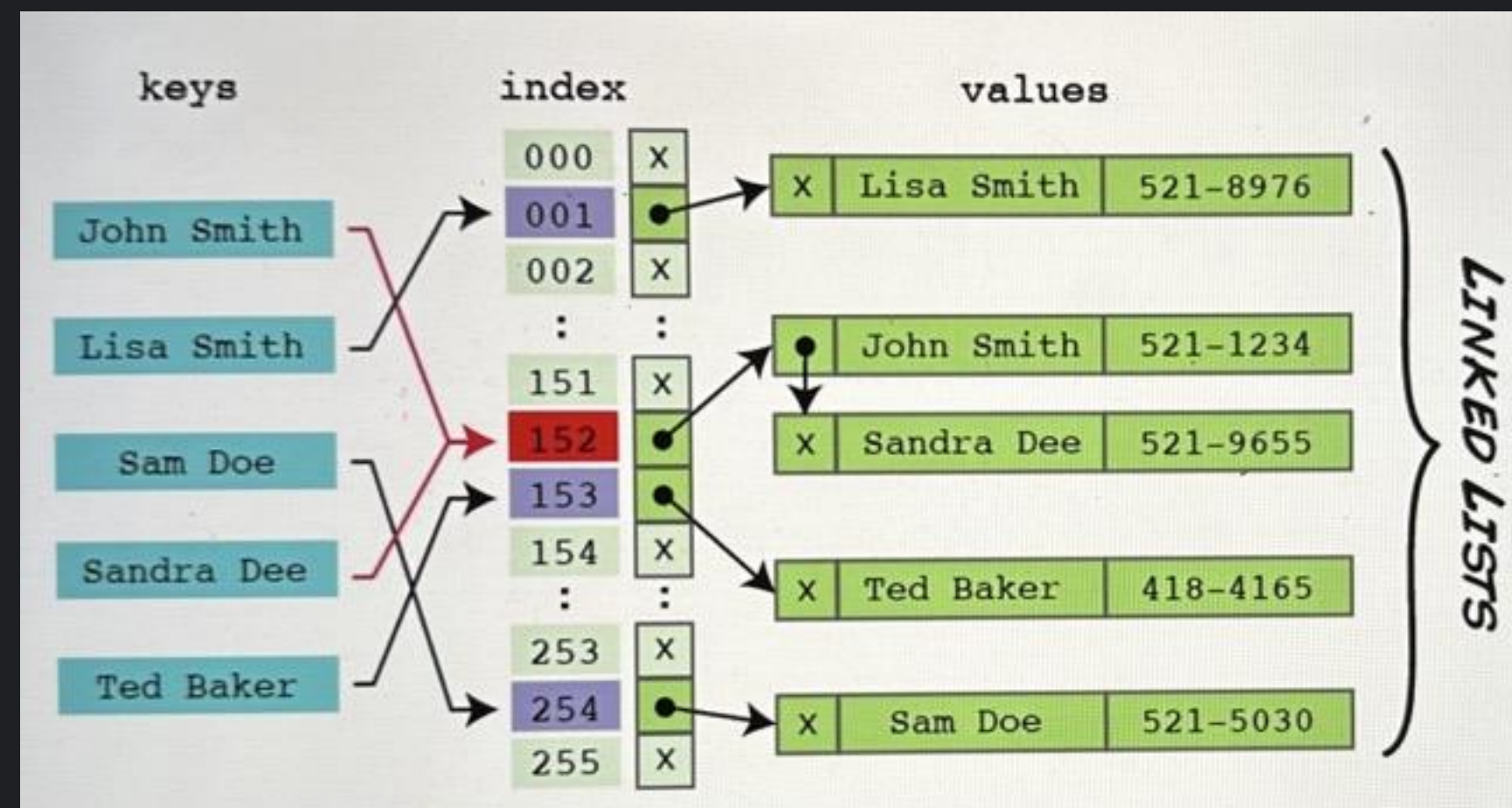
aabbcsabca
aabbssaca

Давайте попробуем посчитать Z-функцию с помощью хешей и бинпоиска

Хеш-таблица

Знакомая нам структура, которая работает на хешах.

В нашей реализации создадим много связанных списков и будем раскидывать по ним элементы с помощью хешей, по аналогии того как это делают при балансировке нагрузки



Всем спасибо!

