Хеши

Асатрян Георгий

4 января 2023 г.

Модульная арифметика

Определение

$a \equiv b \mod N \Leftrightarrow \exists k \in \mathbb{Z} | a = b + kN$

Св-ва

$$\begin{cases} a \equiv b \\ c \equiv d \end{cases} \Rightarrow \begin{cases} a + c \equiv b + d \\ ac \equiv bd \end{cases}$$

$$a^b \ ? \ c^d$$

$$\frac{a}{b} \ ? \ \frac{c}{d}$$

Сравнение строк

Лексикографический порядок строк

aaba
abacaba
abada
abadaba
baba

Определение

Строка лексикографически меньше другой, если она в нее строго вложена, либо если первый отличающийся символ раньше встречается в алфавите, чем ему соответствующий.

Как сравнивать?

Искать первый отличающийся символ долго, в худшем случае работает за $O(|s|)$ - длину строки.

Хеш-функция

Различных строк много - строк длины 10 - 26^{10}. Длины 1000 - ...
Хотим уменьшить множество рассматриваемых значений, например до чисел от 1 до m.

Определение

Хеш-функция - это *какая-то* функция, которая по данной строке получает число. Причем для одной и той же строки всегда одно и то же число. Например - длина строки.

Какая?

Обладает свойствами:

- Необратимости
- Лавинообразности
- Стойкости к коллизиям первого рода? (см. Википедия)
- Стойкости к коллизиям второго рода

Парадокс дней рождения

Если мы сгенерируем k различных строк и посчитаем их хеш функции, которые все принимают значения от 1 до m, какова вероятность, что все хеши различны?

$$\left(1 - \frac{1}{m}\right)^{k \cdot (k-1)} \sim 1 - k \cdot (k-1)\frac{1}{m} \sim 1 - \frac{k^2}{m}$$

В некоторых задачах, k может достигать $1e7$, а то и $1e8$, при этом у задачи может быть скажем 100 тестов, и мы хотим, чтобы задача заходила с вероятностью 99%.

$$\left(1 - \frac{k^2}{m}\right)^{100} \geq 1 - \frac{1}{100} \sim \frac{100k^2}{m} \leq \frac{1}{100} \Leftrightarrow m \geq 1e4k^2$$

Полиномиальные хеши

Зафиксируем p, m - простые числа (на самом деле не обязательно).
При этом p больше размера алфавита (а лучше сильно больше).

Определение 1

$$h_1(s) = s_0 + p \cdot s_1 + \cdots + p^{n-1} \cdot s_{n-1} \mod m$$

Определение 2 <– пользуемся им

$$h_2(s) = p^{n-1} \cdot s_0 + p^{n-2} \cdot s_1 + \cdots + s_{n-1} \mod m$$

Св-ва

$$h_2(s|t) = p^{|t|} \cdot h_2(s) + h_2(t) \mod m$$

$$h_2(s[a; b)) = h_2(s[0; b)) - p^{b-a} \cdot h_2(s[0; a)) \mod m$$

Сравнение строк, маленький модуль

Равны ли две строки? Сравним их хеши, если хеши равны, то будем надеяться что и строки равны (это не гарантируется). Если не равны (в частности мы точно знаем, что они не равны, если их длинны различны) и мы хотим найти какая из них меньше, то совершим двоичный поиск(где здесь монотонность?).
Мы выяснили, что $m = 1e9$ не достаточно, а как лучше? Делать больший модуль сложно из-за ограничений языков (про переполнение далее), поэтому лучшее, что мы можем сделать, это посчитать несколько хешей (можно взять тот же p но другое m и мы $\sim$ возвели в квадрат количество различных значений, когда совсем все плохо, можно еще и по третьему модулю посчитать)

Хорошие и плохие практики

- Модуль $1e9 + 7, 1e9 + 9$ - плохой модуль. Модуль 2^{64} - плохой модуль. Любой другой простой модуль $\sim 1e9$ обычно хороший модуль.
- Когда ты хочешь заменить хеш на двойной(тройной) хеш, идти по всему коду и добавлять вторую строчку - очень плохо, вынести подсчет хешей в отдельный класс/ функции - хорошо.

Плюсы, переполнение, взятие по модулю

.