

Хеширование

Минский ШАД. Осень

17 января 2015 г.

1 Обозначения

В данной домашней работе будет много задач на строки. Введём следующие обозначения:

- $|s|$ — длина строки s
- $s[i]$ — i -й символ строки s
- $s[i \dots j]$ — подстрока строки s , которая начинается в индексе i и заканчивается в индексе j
- \bar{s} — «перевёрнутая» строка s
- $\text{ord}(c)$ — произвольная инъективная функция из алфавита строки в целые числа. Тут будем считать, что символы пронумерованы по алфавиту, т.е. $\text{ord}(a) = 1, \text{ord}(b) = 2, \dots$

Например, если $s = \text{«abacaba»}$, то:

- $|s| = 7$
- $s[3] = \text{'s'}$
- $s[1 \dots 3] = \text{«bac»}$
- $\overline{s[1 \dots 3]} = \text{«cab»}$

2 Тематические задачи

1. Предложить, как решать с помощью полиномиального хеширования следующие задачи (везде вам дана строка s , причём $|s| = n$):
 - (a) $[1/2 \text{ балла}]$ По данным парам (l_1, r_1) и (l_2, r_2) отвечать на запрос, правда ли, что равны две строки $s[l_1 \dots r_1]$ и $s[l_2 \dots r_2]$ за $\mathcal{O}(1)$. Разрешается делать препроцесс за $\mathcal{O}(n)$
 - (b) $[1/2 \text{ балла}]$ По данным парам (l_1, r_1) и (l_2, r_2) отвечать на запрос, правда ли, что равны две строки $s[l_1 \dots r_1]$ и $\overline{s[l_2 \dots r_2]}$ за $\mathcal{O}(1)$. Разрешается делать препроцесс за $\mathcal{O}(n)$
 - (c) $[1/2 \text{ балла}]$ По данной паре (l, r) отвечать на запрос, правда ли, что строка $s[l \dots r]$ является палиндромом за $\mathcal{O}(1)$. Разрешается делать препроцесс за $\mathcal{O}(n)$
 - (d) $[1/2 \text{ балла}]$ Найти по данным (i, j) длину наибольшего общего префикса двух строк $s[i \dots |s|]$ и $s[j \dots |s|]$ за $\mathcal{O}(\log n)$. Разрешается препроцесс за $\mathcal{O}(n)$
 - (e) $[1/2 \text{ балла}]$ Вычислить z -функцию строки за $\mathcal{O}(n \log n)$ (т.е. найти z_i для всех $i = \overline{1 \dots |s|}$). z_i — длина наидлиннейшей подстроки, которая начинается в символе с индексом i и совпадает с префиксом строки.
 - (f) $[1/2 \text{ балла}]$ Для пары (i, j) выяснить, какой префикс лексикографически меньше: который начинается в i или который начинается в j . Время работы $\mathcal{O}(\log n)$. Препроцесс за $\mathcal{O}(n)$

- (g) [1 балл] Построить суффиксный массив для строки s за время $\mathcal{O}(n \log^2 n)$. i -суффиксом (suf_i) назовём подстроку $s[i \dots |s|]$. Суффиксный массив a_i — перестановка первых n чисел, такая, что $\text{suf}_{a_i} < \text{suf}_{a_{i+1}}$ для любого $i = 1 \dots |s| - 1$. Сравнение проводится лексикографически.

Решение:

Повторим идею полиномиального хеширования. Рассмотрим строку «abacaba». Зафиксируем число p (для определённости возьмём тройку), модуль N (для определённости возьмём 1234) и каждому индексу i поставим в соответствие число $p^i \text{ord}(s[i]) \bmod N$. Также посчитаем куммулятивный массив таких сумм h (тут тоже суммируем по модулю N):

s_i	a	b	a	c	a	b	a
i	0	1	2	3	4	5	6
$\text{ord}(s_i)$	1	2	1	3	1	2	1
p^i	1	3	9	27	81	243	729
a_i	1	6	9	81	81	486	729
h_i	1	7	16	97	178	664	159

Для удобства записи отождествим s_i и $\text{ord}(s_i)$ в дальнейшем рассуждении.

Пусть нас спросили, равны ли подстроки $s[1 \dots 2]$ и $s[5 \dots 6]$. Поступим так же, как обычно поступают с куммулятивным массивом, т.е. попробуем посчитать сумму на подотрезке. Что мы получим для первой подстроки: $h_2 - h_0 = ps_1 + p^2s_2 = 15$, для второй: $h_6 - h_4 = p^5s_5 + p^6s_6 = [Вычисления по модулю] = 1215$.

Заметим, что полученные выражения отличаются лишь в показателях степеней. Естественный выход — домножить одну величину на «разность» между степенями, а именно $(h_2 - h_0)p^{5-1} = 15 \cdot 81 = 1215 = h_6 - h_4$. Т.е. хеши действительно совпали.

Таким образом, общий метод для проверки подстрок на совпадение:

- Проверить, что $r_1 - l_1 = r_2 - l_2$, иначе строки сразу не равны
- Пусть, не теряя общности, $l_1 \leq l_2$. Тогда проверим на равенство $(h_{r_1} - h_{l_1-1})p^{l_2-l_1}$ и $h_{r_2} - h_{l_2}$. Если хеши не совпали, то строки различны. Иначе можно с высокой долей уверенности говорить, что строки тоже совпадают.

Для пункта b кроме хешей для строки s предприсчитаем хеши для строки \bar{s} . Теперь надо просто находить разности из двух разных куммулятивных массивов.

Для пункта c воспользуемся решением предыдущего пункта (положим $l_1 = l_2$, $r_1 = r_2$)

Для пункта d воспользуемся бинарным поиском по ответу, а именно будем перебирать длину этой подстроки и сравнивать подстроки описанным методом (пусть текущее значение длины строки — k , тогда можно положить $l_1 = i$, $r_1 = i + k - 1$, $l_2 = j$, $r_2 = j + k - 1$).

В пункте e надо лишь применить решение из d, считая, что $j = 0$.

В пункте f найдём длину самой длинной подстроки, i -префикса и j -префикс (пункт d), а затем сравним первый несовпадающий символ двух префиксов (просто посмотрев в строку на соответствующие места).

В пункте g просто отсортируем массив из первых n чисел с помощью компаратора, описанного в пункте f.

- Предложить функцию для хеширования мультимножеств. А именно, по мультимножеству A и числу m ваша функция $h(A, m)$ должна выдавать число в диапазоне $0 \dots 2^m - 1$, такое что (можно считать, что у вас есть хеш-функция для любого возможного элемента мультимножества, которая вычисляется за $\mathcal{O}(1)$):

- (a) [$\frac{1}{2}$ балла] $h(A, m) = h(B, m)$, если $A = B$
- (b) [$\frac{1}{2}$ балла] Функция должна быть легко обновляемая (т.е. при добавлении элемента в мультимножество должно быть можно пересчитать значение $h(A \cup \{x\}, m)$ за $\bar{o}(|A|^\varepsilon)$, для любого $\varepsilon > 0$)
- (c) [1 балл] Функция должна быть суръективна (можно считать, что функция хеширования элемента суръективна)
- (d) [3 балла] Функция должна быть стойкой. С целью упрощения будем считать, что функция стойкая, если выполняется хотя бы одно из двух:
- Рассмотрим конечное множество элементов B и будем считать, что все элементы мультимножества лежат в B . Зададимся числом n и рассмотрим множество мультимножеств $S_n = \{A : |A| \leq n\}$. Функцию будем называть стойкой, если $\forall m$ и для любого k ($0 \leq k < 2^m$), $P\{h(A, m) = k | A \in S_n\} \rightarrow \frac{1}{2^m}$, при $n \rightarrow \infty$
 - Функцию будем называть стойкой, если для достаточного большого m и $|A| \nmid$ такая константа k , что $\forall |A| \exists C, D$, такие что $|C| \leq k$, $|D| \leq k$ и $h(A, m) = h((A \cup C) \setminus D, m)$

Решение:

Для пункта а и b подходит, например $h \equiv 0$.

Для пункта с можно использовать, например, $h(A) = \bigoplus_{x \in A} H(x)$

Для пункта d нужно использовать уже хорошие функции. Предыдущая функция не подходит, потому что можно дважды добавить один и тот же элемент и получить тот же хеш. Хорошей функцией может являться, например, такая $h(A) = \prod_{x \in \text{uniq}(A)} H(x)^{\text{cnt}(x)}$, где cnt — кратность элемента x в A , а uniq — множество всех элементов мультимножества. За доказательством этого факта можно обратиться сюда.

3 Задачи на повторение

3. [1 балл] Задано прямоугольное поле размерами $n \times m$. В клетке с координатами (i, j) находится ровно $a_{i,j} \in \mathbb{N}$ котиков. Нужно найти такой путь из клетки $(1, 1)$ в клетку (n, m) , что суммарное количество котиков на пути будет максимально. Двигаться по пути можно только вправо, либо только вниз (т.е. разрешённые ходы $(i, j) \rightarrow (i + 1, j)$ либо $(i, j) \rightarrow (i, j + 1)$). Время работы должно составлять $\mathcal{O}(nm)$
4. Задано поле, как в предыдущей задаче. Путь из клетки $(1, 1)$ в клетку (n, m) , двигаясь только вправо либо вниз, назовём «путём сильной и независимой женщины». По пути сильная независимая женщина может в каждой клетке взять или не взять ровно одного кота (т.е. из одной клетки можно взять только одного, но можно брать хоть в каждой клетке пути). Надо сказать, сколько минимум нужно сильных независимых женщин, чтоб собрать всех котов с поля.
- (a) [1 балл] Время решения должно быть $\mathcal{O}(nm \times \sum a_{i,j})$
- (b) [2 балла] Время решения должно быть $\mathcal{O}(nm)$

4 Практические задачи

5. [1 балл] Реализуйте решение задачи ?? (<http://unexisting/link/to/contest.yandex.ru>).
6. [1 балл] Реализуйте задачу бинаризации корневого дерева (<http://second/unexisting/link/to/contest.yandex.ru>).

Задание	1	2	3	4	5	6	Сумма
Баллы	4	5	1	3	1	1	15