SPb HSE, ПАДИИ, 1 курс, весна 2023/24
Практика по алгоритмам #13

Строки-2
23 апреля

Собрано 30 апреля 2024 г. в 14:14

Содержание

1. Строки-2 **1**

2. Разбор задач практики **2**

3. Домашнее задание **4**

3.1. Дополнительная часть 4

Строки-2

1. **Наибольшая дважды подстрока**

   Найти наибольшую по длине строку, которая дважды без перекрытий встречается в заданной строке. $\mathcal{O}(n \log n)$.

2. **LCP[i,j]**

3. Найдите период строки с помощью Z-функции.

4. Количество различных подстрок с помощью Z-функции.

5. **Префикс-функция.**

   a) Определение: $\pi_i = \max j < i \colon s[0, j) = s(i-j, i]$.
   b) Найдите период строки с помощью префикс-функции.
   c) Количество различных подстрок с помощью префикс-функции.
   d) Для каждого префикса число различных подстрок.

6. Поиск с одной ошибкой за $\mathcal{O}(|s| + |t|)$.

7. **Поиск с перестановками символов и алфавита**

   Найти образец в строке, если допустимо:

   a) в образце переставлять символы.
   b) в образце переставлять и алфавит, и символы.
   c) в образце применять к алфавиту перестановку.

8. **Хеширование множеств**

   С множествами могут делать операции: `add(x)`, `del(x)`, `(*) increaseAll(`Δ`)`.
   Нужно уметь сравнивать множества за $\mathcal{O}(1)$ на равенство.

Разбор задач практики

1. **Наибольшая дважды подстрока**

   Бинарный поиск по ответу. Внутри бинпоиска для каждого хеша подстроки длины x в `unordered_map` запоминаем самое левое и самое правое вхождения подстроки.

2. **LCP[i,j]**

   Динамика. $LCP[i,j] = 1 + LCP[i{+}1, j{+}1]$ или 0, если первый символ не совпал.

3. **Найдите период строки с помощью Z-функции.**

   Проверка для периода t: $z[t] = n{-}t$.

4. **Количество различных подстрок с помощью Z-функции.**

   $\forall i$ насчитаем z от суффикса $s[i{:}n)$ и прибавим к ответу $n - i - \max z_j$, т.е. те префиксы i-го суффикса, которые не встречаются правее.

5. **Префикс-функция.**

   a) Определение: $\pi_i = \max j < i\colon s[0,j) = s(i{-}j, i]$.
   b) Ответ: $n - \pi[n{-}1]$
   c) Индукция по длине строки, можно добавлять в конец, можно в начало.
   d) Нам говорят «научитесь добавлять символ в конец». Пусть $\pi_j[i]$ – префикс функция для суффикса $s[j{:}n)$, тогда $f[i{+}1] - f[i] = (n{-}i) - \max_j \max_{p>i} \pi_j[p]$, где $(n{-}i)$ – общее число новых строк, а вычли мы те, что уже встречались ранее.

6. **Поиск с одной ошибкой**

   Ищем s в t. Переберём начало вхождения i. Чтобы проверить i за $\mathcal{O}(1)$,
   хотим узнать LCP суффикса $t[i:]$ и строки s, это ровно $z(s\#t)[i + |s| + 1]$ (z-функция).
   После LCP совпадений идёт или конец строки, или ошибка. Осталось проверить равенство куска строки после ошибки: или хеши, или посмотреть на $z(\overline{s}\#\overline{t})$.

7. **Поиск с перестановками символов и алфавита**

   Найти образец в строке, если допустимо:

   a) в образце переставлять символы.
   b) в образце переставлять и алфавит, и символы.
   c) в образце применять к алфавиту перестановку.

8. **Хеширование множеств**

Мы можем поддерживать для множества A величину $\sum_{a\in A} f(a) \bmod m$.

Осталось придумать f: легко посчитать и сложно подделать. $f(a) = P^a \bmod m$.
$\forall$`a` `a+=`Δ`a` $\Leftrightarrow f(a)$`*=`$P^{\Delta\texttt{a}} \bmod m \Rightarrow$ все три операции `add`, `del`, `+=` за $\mathcal{O}(\log a)$.
Вероятность ошибки: P случайное и должно быть корнем многочлена $\deg = \max a \Rightarrow$
$\Pr \leqslant \frac{\max a}{m}$. $\max a \leqslant 10^9, m \leqslant 10^{18} \Rightarrow$ `OK`. Но для $A_1 = \{0\}$, $A_2 = \{m-1\}$ хеши совпадут $\forall P$.

Мемное решение. Пусть $f(x) =$ `mem`$[x]$, где `mem` — хеш-таблица, куда при первом обращении кладётся псевдорандом. $\Pr$[ошибки] $= \frac{1}{m}$, время `add`, `del` $\mathcal{O}(1)$, но много лишней памяти.

Третий вариант. $A = \prod_{a\in A}(z-a) \bmod m$, где z фиксированное случайное.
Множества A и B имеют коллизию $\Leftrightarrow z$ — корень многочлена степени $\leqslant \max|A|, |B| \Rightarrow$
$\Pr$[ошибки] $\leqslant \frac{|A|}{m}$. `add` за $\mathcal{O}(1)$, `del` за $\mathcal{O}(\log m)$ т.к. нужно обращать по модулю.

Итого. Первое — единственное решение с `+=`. Хеш-таблица даёт самое быстрое решение, но ест много памяти. Третье решение лучше первого по ошибке и времени `add`.

Домашнее задание

1. **(2) Поиск с двумя ошибками**

   Даны две строки – s и t. Нужно найти отрезок t, равный s. При сравнении строк, если несовпадений символов было не более двух, строки считаются равными. $\mathcal{O}(|s| + |t|)$.

2. **(2) Ретростока**

   Для каждого префикса строки найти количество его префиксов равных его суффиксу. $\mathcal{O}(n)$.

3.1. Дополнительная часть

1. **(3) Покрытие строки**

   Говорят, что строка α покрывает строку s, если каждый символ s покрыт хотя бы одним вхождением α. Иначе говоря, все вхождения α в s, как отрезки, покрывают всю s. Дана s, найти минимальную по длине α, покрывающую s. $\mathcal{O}(n \log n)$.