字符串算法入门

realskc

宁波市镇海中学

2024 年 2 月 19 日

定义

字符串是由若干种字符连接而成的串。

由一些字符组成的集合我们称为字符集。常见的字符有大小写拉丁字母、数字等等。

常见的字符集有小写拉丁字母、数字、'01'、'ATGC' 等等。

字符集的种类往往与我们解题无关，我们只需要在乎字符集内的字典序，字符集的大小。

字符串的表示

全课件使用 Python 表示法，具体的，会用到以下几种表示：

$len(s)$，为 s 的长度，即字符个数（也会用 $|s|, n$ 表示），$s[l]$ 为 s 的第 $l+1$ 个字符。

$s[l:r]$ 为 s 的子串，即第 $l+1$ 到第 r 个字符组成的字符串。

$s[l:r:d]$ 为 $s[l:r]$ 每 d 个字符取一个字符（第一个字符取），事实上我们允许 $d<0$，但这里只需记得 $s[::-1]$ 为 s 的翻转。

特别的，上述所有内容中，如果 $l<0$，则将其加上 $len(s)$，r 同。

l 和 r 可以省略不写，但是：必须保留，l 默认为 0，r 默认为 $len(s)$（好像 $d<0$ 不是，但不用管）。

要讲字符串，就要讲哈希。

要讲字符串，就要讲哈希。
要讲哈希，就不能只讲字符串哈希。所以后面会有一些非字符串题。

1 前言

2 哈希
- 字符串哈希
- 其它哈希
- 树哈希

3 单模式串匹配

4 多模式串匹配

5 回文字符串结构

6 后缀字符串结构

P3370 【模板】字符串哈希

如题，给定 N 个字符串（第 i 个字符串长度为 M_i，字符串内包含数字、大小写字母，大小写敏感），请求出 N 个字符串中共有多少个不同的字符串。

$N \leq 10000$，$M_i \approx 1000$，$Mmax \leq 1500$。

字符串哈希

我们最常见的 Hash 为：

$$hash(s) = \sum_{i=0}^{len(s)-1} s[i] * base^i \pmod{P}$$

如果两个字符串的 hash 值相同，我们视为相同，如果 hash 值不同，那么肯定不同。

如果我们在 $[0, P)$ 中均匀随机选取 $base$，一次询问的错误率上界为 $O(\frac{n}{P})$。

如果我们在 $[0, P)$ 中均匀随机选取 $base$，一次询问的错误率上界为 $O(\frac{n}{P})$。

字符集无限大时，容易构造错误率为 $O(\frac{n}{P})$ 的数据。

如果我们在 $[0, P)$ 中均匀随机选取 $base$，一次询问的错误率上界为 $O(\frac{n}{P})$。

字符集无限大时，容易构造错误率为 $O(\frac{n}{P})$ 的数据。

字符集不太大时，如果存在 $d \mid P-1$ 且 $d \leq n$，则可以卡到 $O(\frac{d}{P})$。因此不要用 998244353 作为模数。

如果我们在 $[0, P)$ 中均匀随机选取 $base$，一次询问的错误率上界为 $O(\frac{n}{P})$。

字符集无限大时，容易构造错误率为 $O(\frac{n}{P})$ 的数据。

字符集不太大时，如果存在 $d \mid P-1$ 且 $d \leq n$，则可以卡到 $O(\frac{d}{P})$。因此不要用 998244353 作为模数。

随机比较一个字符串的两个字串等复杂情况，至多只能卡到单次 $O(\frac{\log n}{P})$。

如果我们在 $[0, P)$ 中均匀随机选取 $base$，一次询问的错误率上界为 $O(\frac{n}{P})$。

字符集无限大时，容易构造错误率为 $O(\frac{n}{P})$ 的数据。

字符集不太大时，如果存在 $d \mid P-1$ 且 $d \leq n$，则可以卡到 $O(\frac{d}{P})$。因此不要用 998244353 作为模数。

随机比较一个字符串的两个字串等复杂情况，至多只能卡到单次 $O(\frac{\log n}{P})$。

一般来说 OI 中的哈希都是卡不到 $O(\frac{n}{P})$ 的，视为 $O(\frac{\log n}{P})$ 即可。如果随机模数则更不可能卡。

如果我们在 $[0, P)$ 中均匀随机选取 $base$，一次询问的错误率上界为 $O(\frac{n}{P})$。

字符集无限大时，容易构造错误率为 $O(\frac{n}{P})$ 的数据。

字符集不太大时，如果存在 $d \mid P-1$ 且 $d \leq n$，则可以卡到 $O(\frac{d}{P})$。因此不要用 998244353 作为模数。

随机比较一个字符串的两个字串等复杂情况，至多只能卡到单次 $O(\frac{\log n}{P})$。

一般来说 OI 中的哈希都是卡不到 $O(\frac{n}{P})$ 的，视为 $O(\frac{\log n}{P})$ 即可。如果随机模数则更不可能卡。

使用哈希时要评估正确率，以便确定合适的模数大小。

二分哈希求 LCP

哈希可以 $O(1)$ 比较两个子串是否相同，因此求两个串的最长公共前缀可以先二分长度，再用哈希判断，复杂度 $O(\log n)$。

生日悖论

给定长度为 n 的字符串，查询其 q 个子串中有多少种本质不同的子串。

生日悖论

给定长度为 n 的字符串，查询其 q 个子串中有多少种本质不同的子串。

错误率等效于 $O(q^2)$ 次单次比较。

卡自然溢出

有部分多项式在自然溢出意义下恒为 0，例如 $x^{64}(x+1)^{64}$ 和 $x(x+1)(x+2)\cdots(x+65)$。

卡自然溢出

有部分多项式在自然溢出意义下恒为 0，例如 $x^{64}(x+1)^{64}$ 和 $x(x+1)(x+2)\cdots(x+65)$。

但在字符集不大时，这多项式系数过大无法用于卡哈希。

卡自然溢出

有部分多项式在自然溢出意义下恒为 0，例如 $x^{64}(x+1)^{64}$ 和 $x(x+1)(x+2)\cdots(x+65)$。

但在字符集不大时，这多项式系数过大无法用于卡哈希。

考虑 $(x-1)(x^2-1)(x^4-1)(x^8-1)\cdots(x^{512}-1)$，它具有相同的性质，且系数均为 1 或 -1。它的 i 次项系数为 $(-1)^{\text{__builtin_parity(511-i)}}$。

1 前言

2 哈希
- 字符串哈希
- 其它哈希
- 树哈希

3 单模式串匹配

4 多模式串匹配

5 回文字符串结构

6 后缀字符串结构

哈希适用于 YES 的条件十分苛刻的情况，~~换句话说就是不可以总司令。~~

经典哈希套路

如果只关心每种数出现次数的奇偶性，则可以把每种数都重新映射到一个随机数，然后使用异或。

经典哈希套路

如果只关心每种数出现次数的奇偶性，则可以把每种数都重新映射到一个随机数，然后使用异或。

取出少量（k 个）颜色不相同的元素，可以给每种颜色重新分配一个 $1 \sim k$ 的颜色，然后状压。

https://www.luogu.com.cn/blog/skc/random-algorithm-1

SOJ475【SPC #2】美丽的序列

小 ω 定义美丽的数字为在一个区间中，它出现了偶数次；小 ω 又定义了美丽的区间，一个区间是美丽的当且仅当它里面所有出现过的数都是美丽的；然后小 ω 定义了连续序列的美丽值，也就是这个序列中有多少连续子序列是美丽的。

所以小 ω 给出一个序列，求它的美丽值。

但小 ω 觉得这题太水了，于是小 ω 又加了一个多次询问：每次给出一个区间 $[l, r]$，询问原序列的连续子序列 $S[l..r]$ 的美丽值。

$1 \leq N, Q \leq 10^5; S_i \in [1, 10^6]$。

P7450 [THUSCH2017] 巧克力

「人生就像一盒巧克力，你永远不知道吃到的下一块是什么味道。」

明明收到了一大块巧克力，里面有若干小块，排成 n 行 m 列。每一小块都有自己特别的图案，它们有的是海星，有的是贝壳，有的是海螺……其中还有一些因为挤压，已经分辨不出是什么图案了。明明给每一小块巧克力标上了一个美味值 $a_{i,j}$（$0 \leq a_{i,j} \leq 10^6$），这个值越大，表示这一小块巧克力越美味。

正当明明咽了咽口水，准备享用美味时，舟舟神奇地出现了。看到舟舟恳求的目光，明明决定从中选出一些小块与舟舟一同分享。

舟舟希望这些被选出的巧克力是连通的（两块巧克力连通当且仅当他们有公共边），而且这些巧克力要包含至少 k（$1 \leq k \leq 5$）种。而那些被挤压过的巧克力则是不能被选中的。

明明想满足舟舟的愿望，但他又有点「抠」，想将美味尽可能多地留给自己。所以明明希望选出的巧克力块数能够尽可能地少。如果在选出的块数最少的前提下，美味值的中位数（我们定义 n 个数的中位数为第 $\left\lfloor \frac{n+1}{2} \right\rfloor$ 小的数）能够达到最小就更好了。

你能帮帮明明吗?

$$1 \leq n \times m \leq 233$$

[CSP-S 2022] 星战

n 个点 m 条边的有向图，每条边都有激活和失活两种状态，初始时均为激活状态。四种操作：

1. 失活某条边
2. 失活以某个点为终点的所有边
3. 激活某条边
4. 激活以某个点为终点的所有边

然后问：如果只考虑激活的边，是否满足：

- 所有的点出度均为 1
- 所有的点都满足，从这个点出发，可以走到一个环中

1 前言

2 哈希
- 字符串哈希
- 其它哈希
- 树哈希

3 单模式串匹配

4 多模式串匹配

5 回文字符串结构

6 后缀字符串结构

树哈希

树哈希用于判断两棵无编号树是否同构，直接作用于有根树。

判定无根树同构需要先找重心，然后视为有根树。如果重心有两个，可以将其视为两个子树。

树哈希方法参见 https://peehs-moorhsum.blog.uoj.ac/blog/7891。

Prufer 序列

Prufer 序列可以判断两颗有编号无根树是否同构。

Prufer 序列的定义：每次选择一个编号最小的叶节点并删掉它，然后在序列中记录下它连接到的那个节点。重复 $n-2$ 次后就只剩下两个节点，算法结束。

关键性质是，一个点在 Prufer 序列中的出现次数加 1 等于其度数。

图同构

一般的图同构问题没有高效的判定算法，你需要判图同构的时候说明你思路大概率错了。

[NOI2022] 挑战 NPC

给定两棵有根树 G, H 满足 $1 \leq |H| \leq |G| \leq |H| + k$。

可以删除 G 中的若干个节点得到子图 G'。求是否存在一种删除节点的方式，使得删除后得到的子图 G' 满足如下条件：

- G' 连通。
- G' 包含 G 中的根节点。
- G' 和 H 同构。

1 前言

2 哈希

3 单模式串匹配
- KMP
- Z 算法
- 特殊匹配

4 多模式串匹配

5 回文字符串结构

6 后缀字符串结构

border

如果一个 $0 \leq k < n$ 满足 $s[:k] = s[n-k:]$，则其是一个 'border'。

性质一：border 的 border 是 border。

性质二：两个不同的 border 直接也有 border 关系。

性质三：border 事实上呈现一个树状的关系。

可以按顺序求出一个串所有前缀的最长 border。加入下一个字符时，如果 border 能延长则延长，否则对当前 border 的最长 border 进行尝试。

P3375 【模板】KMP

给出两个字符串 s_1 和 s_2，若 s_1 的区间 $[l, r]$ 子串与 s_2 完全相同，则称 s_2 在 s_1 中出现了，其出现位置为 l。

现在请你求出 s_2 在 s_1 中所有出现的位置。

定义一个字符串 s 的 border 为 s 的一个**非** s **本身**的子串 t，满足 t 既是 s 的前缀，又是 s 的后缀。

对于 s_2，你还需要求出对于其每个前缀 s' 的最长 border t' 的长度。

保证 $1 \leq |s_1|, |s_2| \leq 10^6$，$s_1, s_2$ 中均只含大写英文字母。

KMP 算法

称 s_1 为文本串，s_2 为模式串。

对于 s_1 的每个位置，求出从这个位置开始能匹配的最长的 s_2 的前缀的长度。该值可以按顺序求出，加入下一个字符时，如果匹配能延长则延长，否则将匹配缩短至最长 border 继续尝试。

时间复杂度 $O(n)$。

P5829 【模板】失配树

给定一个字符串 s，定义它的 k 前缀 pre_k 为字符串 $s_{1\ldots k}$，k 后缀 suf_k 为字符串 $s_{|s|-k+1\ldots|s|}$，其中 $1 \le k \le |s|$。

有 m 组询问，每组询问给定 p, q，求 s 的 $\boldsymbol{p}$ 前缀和 $\boldsymbol{q}$ 前缀的最长公共 border 的长度。

border 具有一些很牛的性质，最常见的是一个串的所有 border 构成 $O(\log n)$ 个等差数列。

如果我有时间备课的话明天可以讲这个。

[NOI2014] 动物园

T 组数据，每组数据给定一个长为 n 的字符串，对每个 i，求第 i 个前缀的长度不超过 $\lfloor \frac{i}{2} \rfloor$ 的最长 border。

$T \leq 5, n \leq 10^6$。

P3426 [POI2005] SZA-Template

你打算在纸上印一串字母。

为了完成这项工作，你决定刻一个印章。印章每使用一次，就会将印章上的**所有**字母印到纸上。

同一个位置的相同字符可以印多次。例如：用 aba 这个印章可以完成印制 ababa 的工作（中间的 a 被印了两次）。但是，因为印上去的东西不能被抹掉，在同一位置上印不同字符是不允许的。例如：用 aba 这个印章不可以完成印制 abcba 的工作。

因为刻印章是一个不太容易的工作，你希望印章的字符串长度尽可能小。

$n \leq 5 \times 10^5$。

P5287 [HNOI2019] JOJO

初始有一个空串，你需要依次实现 n 个操作，操作共有 2 种：

- 在当前串末尾加入 x 个 c 字符。保证当前串是空串或者串尾字符不是 c。
- 将串复原到第 x 次操作后的样子。

每一次操作后，你都需要将当前的串的所有前缀的最长 border 长度求和并对 998244353 取模输出。

1 前言

2 哈希

3 单模式串匹配
- KMP
- Z 算法
- 特殊匹配

4 多模式串匹配

5 回文字符串结构

6 后缀字符串结构

Z 算法

也叫 exkmp，用于对所有 i 求 $LCP(s[i:], s)$。
具体算法课上讲。

[NOIP2020] 字符串匹配

对于一个字符串 S，求出 S 的所有具有下列形式的拆分方案数：$S = ABC$，$S = ABABC$，$S = ABAB \ldots ABC$，其中 A，B，C 均是非空字符串，且 A 中出现奇数次的字符数量不超过 C 中出现奇数次的字符数量。

$1 \leq T \leq 5$，$1 \leq |S| \leq 2^{20}$

1 前言

2 哈希

3 单模式串匹配
- KMP
- Z 算法
- 特殊匹配

4 多模式串匹配

5 回文字符串结构

6 后缀字符串结构

给定两个字符串 s_1, s_2 和参数 k，字符集大小为 5。

我们认为两个字符串 t_1, t_2 是相似的仅当仅当它们长度相同且它们对应位置不同的字符数不超过 k。

求 s_1 有多少子串与 s_2 是相似的。

$1 \leq |s_1|, |s_2| \leq 5 \times 10^5$，$0 \leq k \leq 5 \times 10^5$。

该问题可以使用卷积解决。

先枚举字符，然后使用卷积计算每个位置处有几个元素是匹配的，对于每种字符求和即可。

SOJ579 【SSR #3】字符串问题

有一个 01 字符串 S，每次小 ω 会修改它，并且会询问一个字符串 S_2，你需要给出 S_2 在 S_1 的一个区间中出现了几次。

具体来说修改方式如下：

1. 区间变为一个字符 v
2. 给定 l，并找出 $S_1 + S_2 = S, |S_1| = l$，并让 S 变为 $S_2 + S_1$
3. 区间异或上 1

$1 \leq T \leq 100, 1 \leq l_i \leq r_i \leq |S| \leq 5000, 0 \leq v_i \leq 1, 1 \leq q \leq 10000$。

如果文本串为 s，模式串为 t，则 bitset 可以在 $O(\frac{|s||t|}{w})$ 的时间内完成匹配。

枚举模式串字符，每个位置是否可匹配容易用文本串得到，对 $|t|$ 个 bitset 取 and 即可。

1 前言

2 哈希

3 单模式串匹配

4 多模式串匹配
- Trie
- AC 自动机

5 回文字符串结构

6 后缀字符串结构

P8306 【模板】字典树

给定 n 个模式串 $s_1, s_2, \ldots, s_n$ 和 q 次询问，每次询问给定一个文本串 t_i，请回答 $s_1 \sim s_n$ 中有多少个字符串 s_j 满足 t_i 是 s_j 的前缀。

$1 \leq T, n, q \leq 10^5$，且输入字符串的总长度不超过 3×10^6。

Trie 是所有模式串构成的最朴素的自动机，自动机自然具有的信息是串的前缀信息。因此 Trie 具有所有模式串的前缀信息。

自动机的定义：`https://oi-wiki.org/string/automaton/`。

由于 Trie 的结构是树形的，因此可以在 Trie 上进行很多操作。

压位 Trie 是一种高效的亚 log 数据结构，可以参见钱哥的论文。

P4551 最长异或路径

给定一棵 n 个点的带权树，结点下标从 1 开始到 n。寻找树中找两个结点，求最长的异或路径。

异或路径指的是指两个结点之间唯一路径上的所有边权的异或。

$1 \leq n \leq 100000; 0 < u, v \leq n; 0 \leq w < 2^{31}$。

SOJ1382 你为什么不用 gedit 写代码呢

开始的时候，你有一个空串。你可以用 $1ms$ 的时间在当前字符串的尾部添加一个字符。如果当前串不是空串，你也可以用 $1ms$ 删除尾部的一个字符。

编辑器中内置了一些关键字，你可以用 $1ms$ 的时间按一下 ‘Tab‘ 来自动补全成关键字。

令这个关键字集合为 S，具体的，如果当前字符串是 x，你想要补全成 $y \in S$，满足 x 是 y 的前缀，则你需要 tms 的时间来进行自动补全，其中 t 是 y 在 S 中所有存在前缀 x 的字符串按照字典序排序的排名。

注意：如果当前字符串 x 为关键字，那么进行第一次（$1ms$）自动补全之后仍会得到 x。

现在给你这个关键字集合，对于每个关键字，请你求出最快需要多少 ms 的时间才能打出来。

[AGC064C] Erase and Divide Game

Takahashi 和 Aoki 玩游戏。先在黑板上写若干个数，由 N 个**互不相交**的区间 $[l_i, r_i]$ 组成。

两人轮流操作，每次操作先删去所有的奇数/偶数，再把剩下的数除以 2（向下取整），无法操作的人输。

Takahashi 先手，假设两人都采用最优策略，问谁能获胜。

1 前言

2 哈希

3 单模式串匹配

4 多模式串匹配
- Trie
- AC 自动机

5 回文字符串结构

6 后缀字符串结构

Trie 作为自动机，只接受模式串。

KMP 可以被更改为自动机，接受所有以模式串结尾的串。

如果一个自动机，接受所有以任何一个模式串结尾的串，则这个自动机被称为 AC 自动机。

P5357 【模板】AC 自动机

给你一个文本串 S 和 n 个模式串 $T_{1\sim n}$，请你分别求出每个模式串 T_i 在 S 中出现的次数。

$1 \leq n \leq 2 \times 10^5$，$T_{1\sim n}$ 的长度总和不超过 2×10^5，S 的长度不超过 2×10^6。

如果场上忘了板子具体怎么写，就思考要维护哪些量，每个量分别怎么求。

SAM 也是一样的道理。

P2444 [POI2000] 病毒

某些确定的二进制串是病毒的代码。如果某段代码中不存在任何一段病毒代码，那么我们就称这段代码是安全的。现在委员会已经找出了所有的病毒代码段，试问，是否存在一个无限长的安全的二进制代码。

示例：例如如果 $\{011, 11, 00000\}$ 为病毒代码段，那么一个可能的无限长安全代码就是 $010101\ldots$。如果 $\{01, 11, 000000\}$ 为病毒代码段，那么就不存在一个无限长的安全代码。

现在给出所有的病毒代码段，判断是否存在无限长的安全代码。

$1 \leq n \leq 2000$，所有病毒代码段的总长度不超过 3×10^4。

P5599 【XR-4】文本编辑器

有一个长度为 n 的文本串 a 和 m 个模式串，第 i 个模式串为 s_i。

- 查找功能：有两个参数 l, r，表示询问对于每个 s_i，$a[l:r]$ 中 s_i 的出现次数之和。
- 替换功能：有三个参数 l, r, t，其中 t 是一个字符串，表示将 $a[l:r]$ 替换为 t 不断重复的结果。即如果把 `Mds72SKsLL` 替换为 `Rabb` 不断重复的结果，则原字符串变为 `RabbRabbRa`。

有 q 个操作，每个操作是查找或替换之一，你需要正确回答每个查找操作的答案。