后缀数组

约定

- 如无特殊说明，均遵守以下约定
- 字符串下标从1开始，字符串S的长度记作$|S|$，默认大小为n
- $S(l,r)$表示S第l个字符到第r个字符形成的子串，若$l > r$则为空串
- “后缀i”、“$suf(i)$”表示以第i个字符开头的后缀$S(i,n)$
- 字符集记作$\sum$，字符集大小为$|\sum|$，例如题目说明字符串只包含小写字母：$|\sum| = 26$

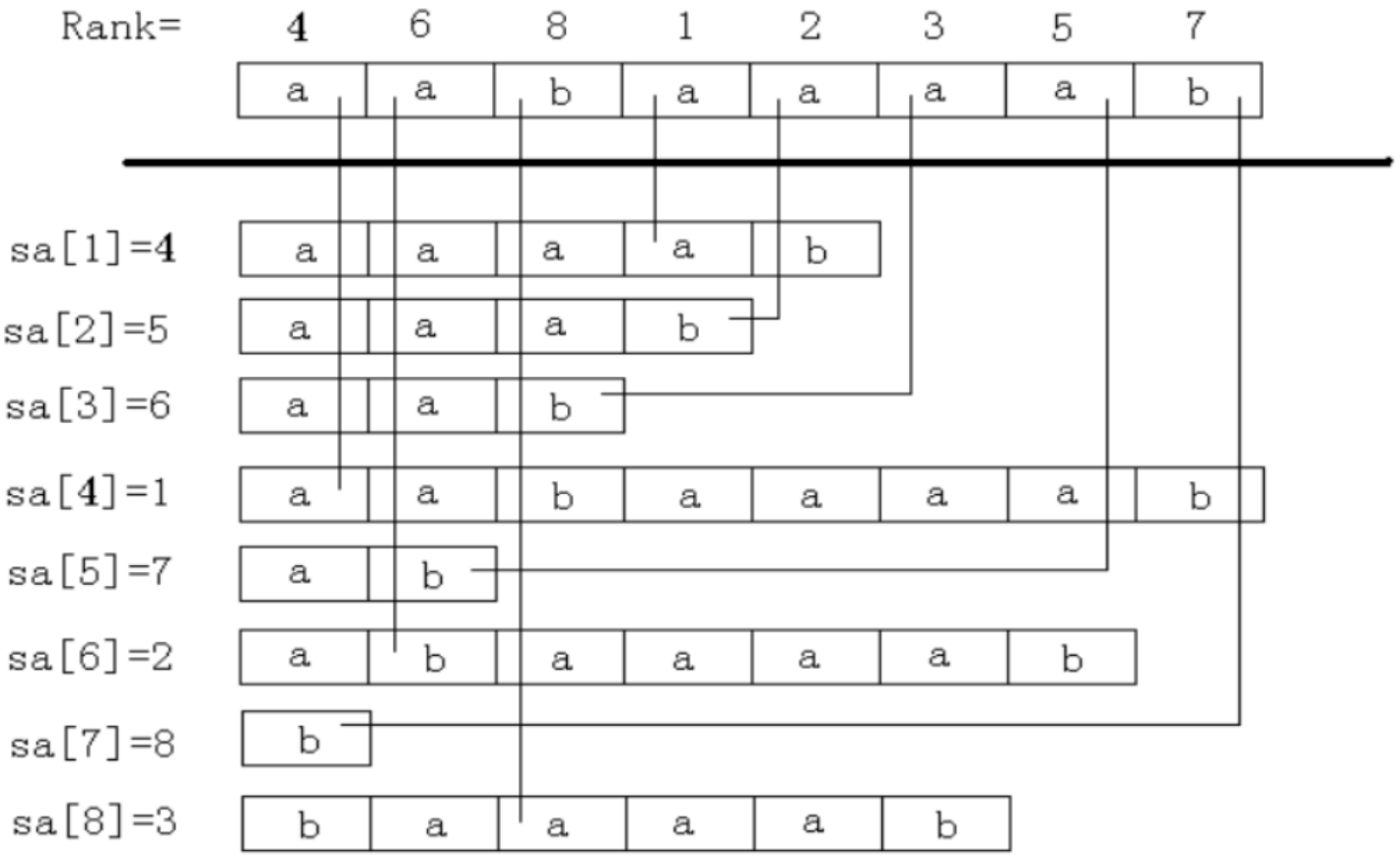
字典序

- 字符串大小的比较，采用逐位比较的方式，从头开始找到第一个不相同的位置，在对应位置上谁更小则整个字符串字典序更小
- abaccccc < abb
- 当字符串长度不同时，可以认为较短的字符串末尾接上了若干个极小的字符，用于补齐
- abb < abba
- 当且仅当两个字符串长度相同且每一位都相同时，字符串相等

后缀数组

- 长度为n的字符串S，共有n个后缀
- $S = abca$有4个后缀$suf(1) = abca$、$suf(2) = bca$、$suf(3) = ca$、$suf(4) = a$
- 后缀数组主要用到两个数组：sa、rk
- $sa[i]$表示将所有后缀排序后第i小的后缀，称为后缀数组
- $rk[i]$表示后缀i的排名，称为排名数组
- 这两个数组满足性质$sa[rk[i]] = rk[sa[i]] = 1$
- 是否存在相等的后缀

后缀数组


Rank= 4 6 8 1 2 3 5 7
a a b a a a a b
sa[1]=4 a a a a b
sa[2]=5 a a a b
sa[3]=6 a a b
sa[4]=1 a a b a a a a b
sa[5]=7 a b
sa[6]=2 a b a a a a b
sa[7]=8 b
sa[8]=3 b a a a a b

后缀数组

- 如何求解?
- 采用最暴力的方法，将带有全部后缀的数组直接sort
- 时间复杂度?
- $O(nlogn)$次比较，每次$O(n)$，总时间复杂度$O(n^2logn)$

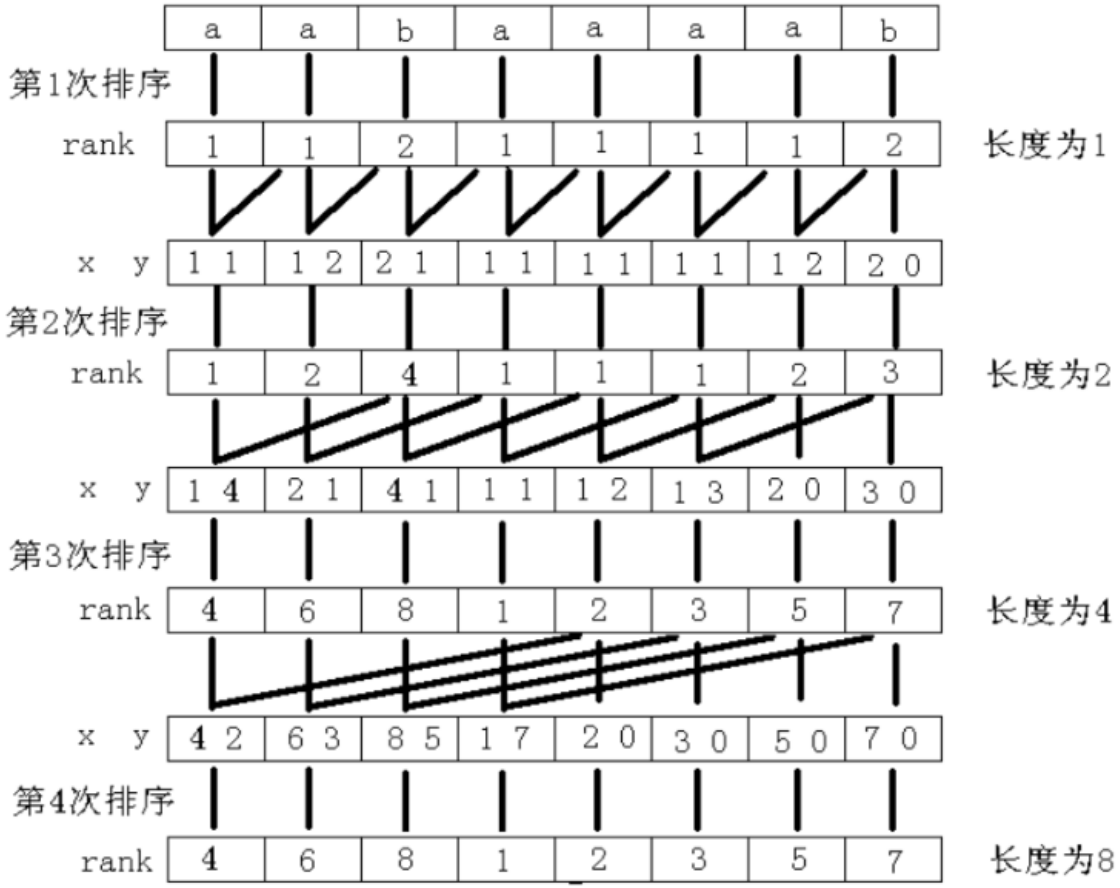
后缀数组

- 设$S = S(1, mid)S(mid + 1, n), T = T(1, mid)T(mid + 1, m)$
- 满足什么条件时能得到$S < T$
- $S(1, mid) < T(1, mid)$
- 或者$S(1, mid) = T(1, mid)$ && $S(mid + 1, n) < T(mid + 1, m)$
- 能否用于刚才的暴力排序?
- 分治/倍增/二分 + hash，单次比较变为$O(logn)$，总时间复杂度$O(nlog^2 n)$
- 虽然不是好的做法，但可以用于救急

后缀数组

- 下面介绍基于倍增的后缀排序
- 在前k轮，将每个$S\left(i, i+2^{k-1}-1\right)$排好序，得到每个串的$rk$
- 在第$k+1$轮时，将$S\left(i, i+2^{k}-1\right)$分为两部分，即
  $$S\left(i, i+2^{k-1}-1\right)S(i+2^{k-1}, i+2^{k}-1)$$
- 利用上一轮的rk，可以将串看作一个pair<int,int>
- 比较的复杂度降为$O(1)$，可以$O(nlogn)$完成一轮排序
- 一共进行$O(logn)$轮，总时间复杂度$O(nlog^2n)$

后缀数组



进行第1轮排序时，每个串仅包含单个字符，可以直接用ASCII码进行比较

当字符串长度不够时，类似之前使用极小字符补齐，这里给pair补0

注意到排序的关键字是排名，值域$O(n)$

将sort改成基数排序，每轮排序仅需$O(n)$，总时间复杂度$O(nlogn)$

后缀数组

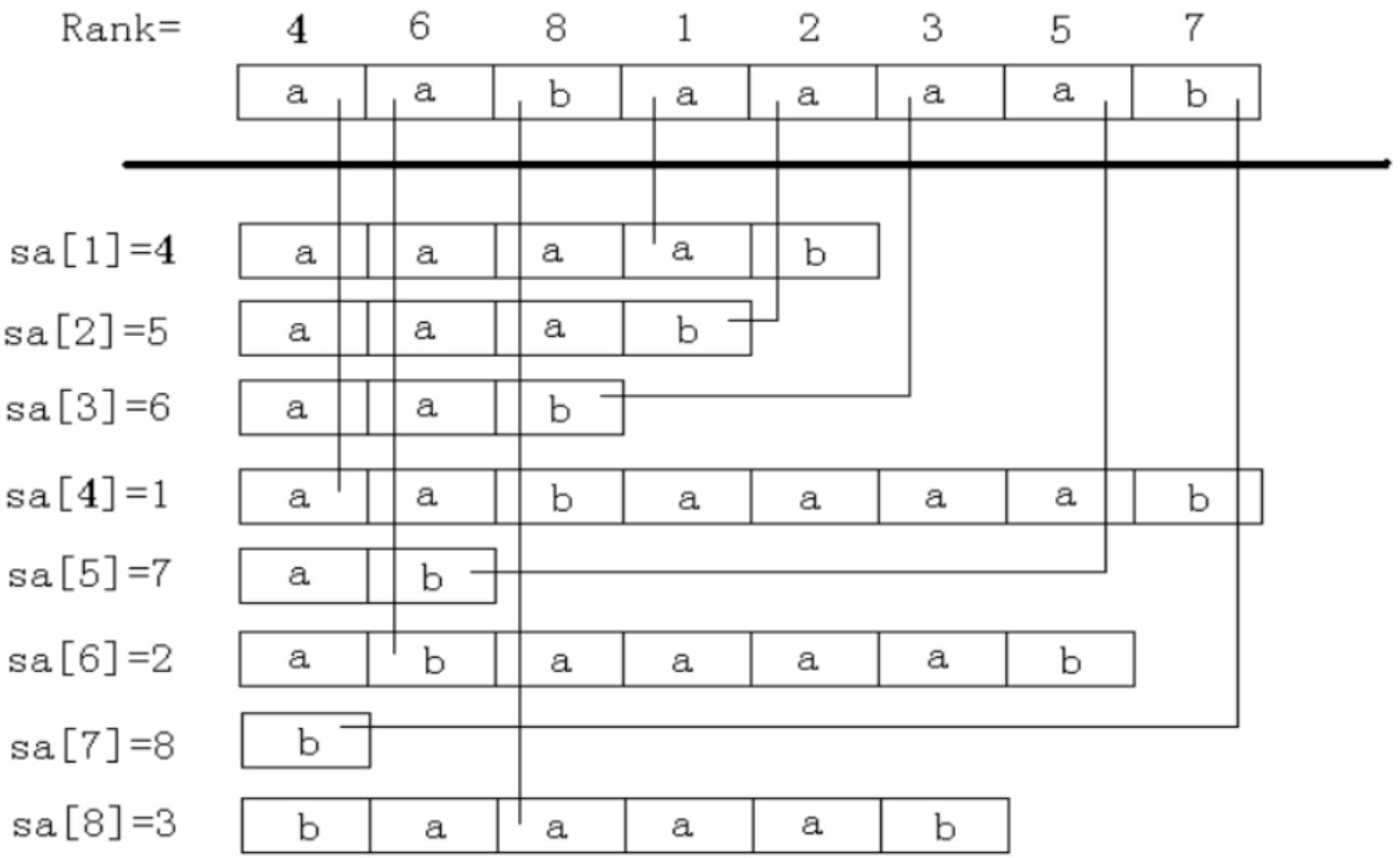
- Infinite Fraction Path
  n个点，下标$0 \ldots n-1$，点i有一条指向$(i^2+1)\%n$的有向边，点上有个字符$a[i]$；初始$S=$“”，每经过一个点x，$S \mathrel{+}= a[x]$；你需要选择一个点，一直沿着边走，直到S的长度为n时停下，求字典序最大的S；

$$n \le 1.5 \times 10^5, |\sum| \le 26$$

- 采用刚才的倍增思想即可，拼接字符串时拼的是走2^k步到达位置的字符串
- 时间复杂度$O(nlogn)$

后缀数组

- $height[i] = lcp(sa[i\ -1],\ sa[i])$



height[1] = 0

height[2] = 3

height[3] = 2

height[4] = 3

height[5] = 1

height[6] = 2

height[7] = 0

height[8] = 1

后缀数组

- 设$h[i] = height[rk[i]]$，有$h[i] \geq h[i-1] - 1$
- 根据结论暴力求解$height$即可，减少不超过n次，增加不超过$2n$次，总时间复杂度$O(n)$
- 求解时按照h的顺序计算，不需要真正的存储h

后缀数组

- $height[i] = lcp(sa[i\ -1],\ sa[i])$
- 求两个子串的最长公共前缀

$$lcp(sa[i], sa[j]) = \min\{height[i+1 \ldots j]\}$$

- 感性的理解一下，如果区间最小值为a，那么前a个字符一直都没有变过，因此lcp至少为a
- 从第$a+1$位开始，因为字符串有序，变走了之后不可能再变回来，因此lcp只能等于a
- 因此求lcp问题转化为了RMQ问题
- 一般使用$O(nlogn) - O(1)$的st表

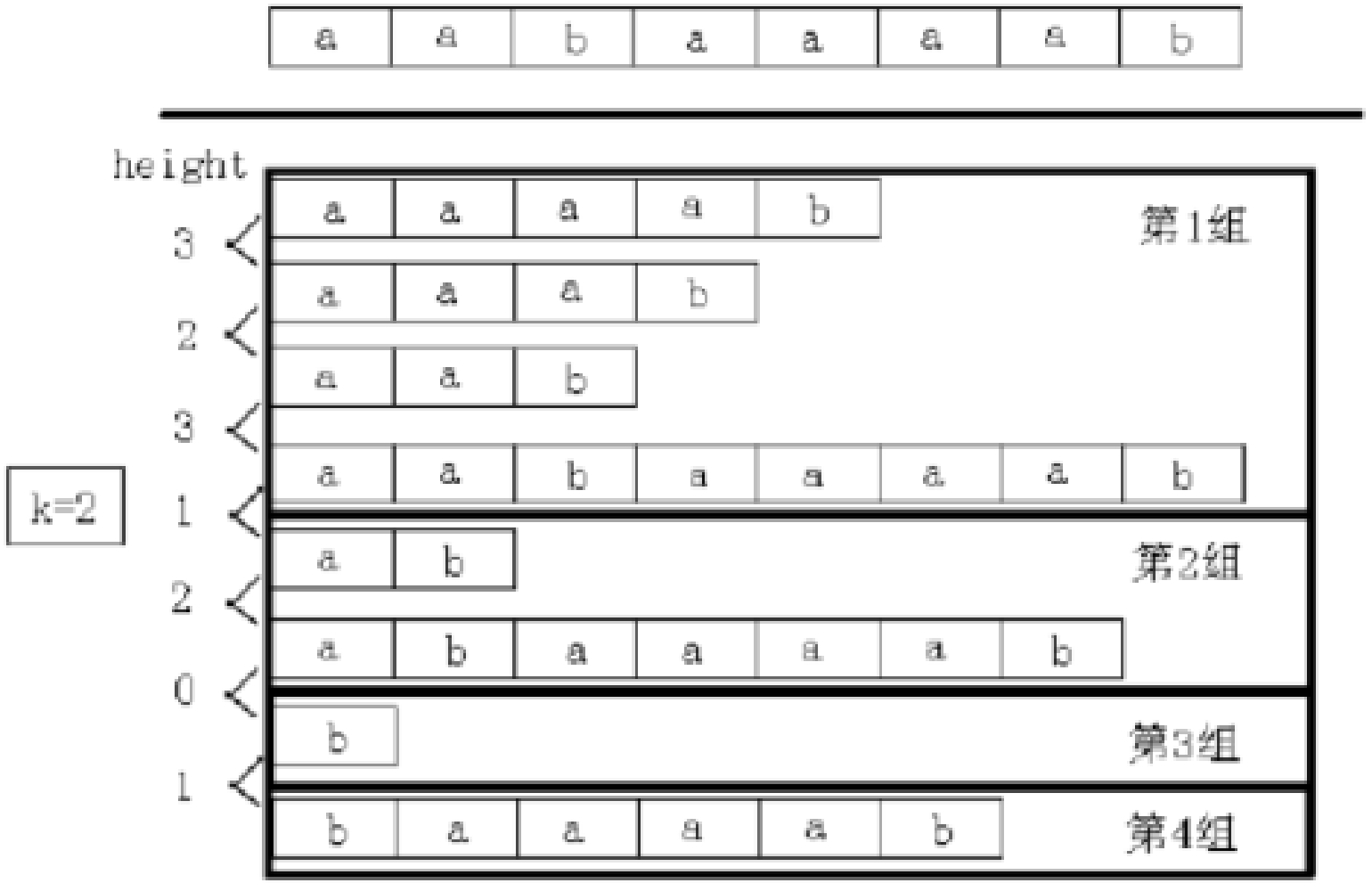
最长重复子串

- 求S出现至少两次的最长子串
- 如何表示一个子串？
- 后缀的前缀
- 出现两次说明可以在两个后缀的前缀同时找到
- 对于同一对后缀(i,j)，只有$lcp(sa[i],sa[j])$可能对答案进行贡献
- 在$height$数组上，找到$[l,r]$，使得区间最小值最大
- 发现区间向两边扩展后值只能变小
- 因此区间长度只能为1

不可重叠最长重复子串

- 求S出现至少两次的最长子串，且子串出现的两个位置不能重叠
- 发现答案具有可二分性，二分子串长度k，转为判定问题
- 如何表示长度为k的子串？
- 对每个满足$height[i] \geq k$的位置，将i和$i-1$合并到一组，这样每一组都表示了一个不同的长度为k的子串
- 如何满足出现位置的限制？
- 组内sa的$Max - Min \geq k$即满足条件

不可重叠最长重复子串


a a b a a a a b
height
3
2
3
1
2
0
1
k=2
a a a a b
a a a b
a a b
a a b a a a a b
a b
a b a a a a b
b
b a a a a b
第1组
第2组
第3组
第4组

Delicious Dessert

- 对于S的子串T，T被称为美味的，当且仅当$cnt(T)\%len(T) == 0$，其中$cnt(T)$表示T在S中的出现次数，求美味的T的个数

$$|S| \leq 10^6, |\sum| \leq 26$$

- 沿用上一道题合并的思路，发现当L变为$L-1$之后，只需要在之前的基础上再合并$height = L-1$的位置
- 块内的每个后缀均代表一个出现位置
- 并查集维护每一组以及siz即可，查询时暴力枚举L的倍数
- 时间复杂度$O(nlogn)$

子串个数

- 求S本质不同子串的个数
- 同样还是数后缀的前缀
- 按照字典序依次考虑每个后缀
- 每次加入一个后缀$sa[i]$时，发现会有一些前缀已经被计入答案了
- 长度在$[height[i]+1, n-sa[i]+1]$之间的前缀对答案有贡献

最长回文子串

- 求S的最长回文子串
- 考虑枚举中点，向两边尽可能拓展
- 似乎出现了前缀的后缀？
- 将S翻转
- 如何求S的后缀和翻转S后得到的后缀之间的lcp？
- 在新串$T = S + '\#' + S^r$上求后缀数组
- 为什么要在中间加一个分隔符？

最小表示

- 给一个字符串S，每次可以将第一个字符移动到最后面，求能得到的字典序最小的字符串
  如$BBAAB$，最小的是$AABBB$
- 构造新串$T = S + S$，答案一定是某个后缀的前缀
- 对T后缀排序，找到第一个长度不小于n的后缀

重复次数最多重复子串

- 求k最大的T^k，满足T^k是S的子串，例如$(ab)^3 = ababab$

$$n \leq 10^5, |\sum| \leq 26$$

- 仅考虑$k \geq 2$的情况
- 枚举T的长度L，在每个$i \times L$的位置设置一个“检查点”，重复出现的子串至少会覆盖两个相邻的检查点
- 两个检查点往前往后尽可能匹配，匹配出来的串一定以L为周期
- 根据总长度可以计算重复出现的次数
- 时间复杂度$O(\sum \frac{n}{i}) = O(nlogn)$

最长公共子串

- 求字符串S和T的最长公共子串
- 同样是找后缀的前缀，只是需要后缀属于不同的串
- 构造新串$S + \text{`\#'} + T$，在上面求后缀数组
- 对于分属S和T的两个后缀，只有lcp可能对答案进行贡献
- 因此同样只能找相邻的

多个串最长公共子串

- 求k个串的最长公共子串，k为小常数
- 将k个串拼起来，中间用分隔符分开
- 二分答案+判定组内是否有分属k个串的后缀
- 观察到区间应该是极小的
- 双指针开扫

AHOI2013 差异

- 给定长度为n的字符串S，求$\sum_{1\le i<j\le n} lcp(suf(i), suf(j))$
  $n \le 5 \times 10^5$
- 注意到$sa[i]$同样是一个排列，且lcp运算可以交换
- 转化为求$\sum_{1\le i<j\le n} lcp(suf(sa[i]), suf(sa[j]))$
- 考虑$height$对答案的贡献，仅当$height$为区间最小值时有贡献
- 利用单调栈求出$height$为区间最小值时向两边拓展的极大区间
- 乘法原理计算贡献即可
- 注意$height$相同时，为避免算重需要钦定大小
- 时间复杂度$O(nlogn)$

K-th occurrence

- 长度为n的串S和m组询问，每次询问(l,r,k)，求$S(l,r)$在S中第k次出现位置

$$n,m \leq 10^5, |\sum| \leq 26$$

- $S(l,r)$是$suf(l)$的前缀，放到后缀数组上考虑
- 向左右$height \geq r-l+1$的位置拓展，区间内每个后缀对应着一次出现，端点可以二分求出
- 问题转化为了求区间k小
- 主席树上查询即可
- $O((n+m)logn)$

HEOI2016/TJOI2016 字符串

- 长度为n的串S和m组询问，每次询问(a,b,c,d)，问$S(a,b)$的所有子串和$S(c,d)$的lcp最大值

$$n,m \leq 10^6, |\sum| \leq 26$$

- $suf(c)$和$S(a,b)$的所有子串求lcp
- $suf(c)$和$[a,b]$中的部分后缀求lcp
- 需要确定长度L才好求
- 答案具有单调性，二分后转化为判定问题
- $O((n+m)logn)$

小结

- 后缀数组主要用到了3个数组sa、rk、height
- LCP问题可以转化为RMQ问题
- 子串可以用后缀的前缀表示，把字符串问题放到后缀数组上去考虑，转化为序列问题
- 可以考虑如果固定长度之后是否好做，枚举/二分
- 枚举长度时可以考虑相邻长度之间的信息是否好转移
- 多个串可以拼接在一起，需要考虑是否需要分隔符，分隔符是否需要不同（fzoj7534）

相关资料

1. https://oi-wiki.org/string/sa/
2. [2004] 后缀数组 by. 徐智磊
3. [2009] 后缀数组——处理字符串的有力工具 by. 罗穗骞