字符串算法选讲

l0nl1f3

福州第三中学

2017年6月

Tips

- 忘了Tips要写什么了

KMP

- 给定一个长度为n字符串S,给定一个长度为m字符串T,问T在S中出现了几次

KMP

- 给定一个长度为n字符串S,给定一个长度为m字符串T,问T在S中出现了几次
- $n \leq m \leq 5 * 10^6$

KMP

- 朴素匹配,枚举起点l,比较$s_{l...l+m-1}$和T是否相等,遇到不等(失配)就移动l
- 复杂度$O(nm)$

KMP

- 朴素匹配,枚举起点l,比较$s_{l...l+m-1}$和T是否相等,遇到不等(失配)就移动l
- 复杂度$O(nm)$
- KMP(Knuth–Morris–Pratt algorithm),俗称看毛片算法

KMP

- 朴素匹配,枚举起点l,比较$s_{l\ldots l+m-1}$和T是否相等,遇到不等(失配)就移动l
- 复杂度$O(nm)$
- KMP(Knuth–Morris–Pratt algorithm),俗称看毛片算法
- 定义一个串的border为满足$s_{1\ldots x} = s_{n-x+1\ldots n}$的前缀

KMP

- 朴素匹配,枚举起点l,比较$s_{l\ldots l+m-1}$和T是否相等,遇到不等(失配)就移动l
- 复杂度$O(nm)$
- KMP(Knuth–Morris–Pratt algorithm),俗称看毛片算法
- 定义一个串的border为满足$s_{1\ldots x} = s_{n-x+1\ldots n}$的前缀
- KMP的next数组存储的是T串每个前缀的最大border的长度

KMP

- 失配时按border跳

KMP

- 失配时按border跳
- 构建next数组$O(n)$，匹配总复杂度$O(n)$

KMP

- 给定一个长度为n字符串S,给定一个长度为m字符串T,问T在S中出现了几次

KMP

- 给定一个长度为n字符串S,给定一个长度为m字符串T,问T在S中出现了几次
- $n \leq m \leq 5 * 10^6$

Easy Period Problem

- 对于一个字符串T,如果存在字符串A,使得$A + A + \ldots (x*A) = T$,其中加法为顺次连接
- 则称$T = A^x$,求满足条件的最短A的长度

Easy Period Problem

- 对于一个字符串T,如果存在字符串A,使得$A + A + \ldots (x*A) = T$,其中加法为顺次连接
- 则称$T = A^x$,求满足条件的最短A的长度
- 设$T = n - next_n$,若$n \bmod T = 0$,则答案为T
- 正确性如何?

Trie

- 给定n个串,第i个串长为l_i,Q次询问(x, y)的最长公共前缀
- $\sum l_i \leq 5 * 10^6, Q \leq 10^6, n \leq 10^5$

Trie

- 给定n个串,第i个串长为l_i,Q次询问(x, y)的最长公共前缀
- $\sum l_i \leq 5 * 10^6, Q \leq 10^6, n \leq 10^5$
- 建立一棵26叉前缀树(trie).每条边上有一个字母
- 根到某个点的路径组成一个前缀
- 每次插入一个字符串时，就在trie树上把对应路径“填满”，并记录末字符所在的节点

Trie

- 给定n个串,第i个串长为l_i,Q次询问(x, y)的最长公共前缀
- $\sum l_i \leq 5 * 10^6, Q \leq 10^6, n \leq 10^5$
- 建立一棵26叉前缀树(trie).每条边上有一个字母
- 根到某个点的路径组成一个前缀
- 每次插入一个字符串时，就在trie树上把对应路径“填满”，并记录末字符所在的节点
- 将公共前缀查询转化为LCA查询
- $O(\sum l_i \log \sum l_i + Q)$

Trie + KMP

- 给定n个串,第i个串长为l_i
- 再给定一个长度为m的文本串,问有多少个串在文本串中出现过
- $\sum l_i \leq 5 * 10^5, m \leq 10^6, n \leq 10^4$

Trie + KMP

- 给定n个串,第i个串长为l_i
- 再给定一个长度为m的文本串,问有多少个串在文本串中出现过
- $\sum l_i \leq 5 * 10^5, m \leq 10^6, n \leq 10^4$
- 大力KMP,复杂度$O(nm)$

Trie + KMP

- Trie + KMP = Aho-Corasick Automation

Trie + KMP

- Trie + KMP = Aho-Corasick Automation
- 对于Trie上的每一个前缀，我们任然可以建立一个像kmp一样的next数组
- 对于Trie进行bfs，考虑将要遍历i节点的c孩子

Trie + KMP

- Trie + KMP = Aho-Corasick Automation
- 对于Trie上的每一个前缀，我们任然可以建立一个像kmp一样的next数组
- 对于Trie进行bfs，考虑将要遍历i节点的c孩子
- 若c孩子不存在,则将孩子指针指向$next_i$的c孩子
- 否则将该孩子的$next$指向$next_i$节点的c孩子

Trie + KMP

- Trie + KMP = Aho-Corasick Automation
- 对于Trie上的每一个前缀，我们任然可以建立一个像kmp一样的next数组
- 对于Trie进行bfs，考虑将要遍历i节点的c孩子
- 若c孩子不存在,则将孩子指针指向$next_i$的c孩子
- 否则将该孩子的$next$指向$next_i$节点的c孩子
- 记录每个点的cnt值为整串结尾在该点串数

Trie + KMP

- 那么我们对文本串进行一个Trie上的kmp即可
- $O(\sum l_i + m)$
- HDU2222

Trie + KMP - Tree

- 给出n个字符串,询问每个字符串在所有字符串中出现的次数之和
- $n \leq 10^5, \sum l_i \leq 10^6$

Trie + KMP - Tree

- 给出n个字符串,询问每个字符串在所有字符串中出现的次数之和
- $n \leq 10^5, \sum l_i \leq 10^6$
- 暴力AC自动机的复杂度???

Trie + KMP - Tree

- 给出n个字符串,询问每个字符串在所有字符串中出现的次数之和
- $n \leq 10^5, \sum l_i \leq 10^6$
- 暴力AC自动机的复杂度???
- $O((\sum l_i)^2)$

Trie + KMP - Tree

- 给出n个字符串,询问每个字符串在所有字符串中出现的次数之和
- $n \leq 10^5, \sum l_i \leq 10^6$
- 暴力AC自动机的复杂度???
- $O((\sum l_i)^2)$
- 我们建立一棵fail树,我们把每个节点x向$next_x$连边

Trie + KMP - Tree

- 给出n个字符串,询问每个字符串在所有字符串中出现的次数之和
- $n \leq 10^5, \sum l_i \leq 10^6$
- 暴力AC自动机的复杂度？？？
- $O((\sum l_i)^2)$
- 我们建立一棵fail树,我们把每个节点x向$next_x$连边
- 每个点都是一个字符串的前缀，而且每个字符串的每个前缀在这棵树上都对应着一个点。
- 其次，由于fail指针，每个点父节点的字符串都是这个点字符串最长的存在的后缀

Trie + KMP - Tree

- 插入字符串时记录每个点被经过的次数
- 最后在fail树上答案体现为子树和
- 复杂度$O(n + \sum l_i)$,BZOJ3172

Manacher

- 给定一个长度为n的字符串S,求S的最长回文子串
- 回文的定义为正反读是一样的,比如qzzq和zqz就是回文,而zzq就不是回文
- $n \leq 10^6$

Manacher

- 给定一个长度为n的字符串S,求S的最长回文子串
- 回文的定义为正反读是一样的,比如qzzq和zqz就是回文,而zzq就不是回文
- $n \leq 10^6$
- 而值得一提的是,(())不是回文,而((((是

Manacher

- 枚举回文中心,暴力扩展,复杂度$O(n^2)$

Manacher

- 枚举回文中心,暴力扩展,复杂度$O(n^2)$
- 设r_i为i为中心的回文半径(偶回文串先不管)
- 添加辅助变量mx和p，分别表示已有回文半径覆盖到的最右边界和对应中心

Manacher

- 枚举回文中心,暴力扩展,复杂度$O(n^2)$
- 设r_i为i为中心的回文半径(偶回文串先不管)
- 添加辅助变量mx和p，分别表示已有回文半径覆盖到的最右边界和对应中心
- i关于p的对称点为$j = 2p - i$,我们进行分类讨论

Manacher

- 对于一个i,若$mx > i$,则$r_i = \min(r_{2p-i}, mx - i)$,否则$r_i = 1$

Manacher

- 对于一个i,若$mx > i$,则$r_i = \min(r_{2p-i}, mx - i)$,否则$r_i = 1$
- 暴力增大r_i

Manacher

- 对于一个i,若$mx > i$,则$r_i = \min(r_{2p-i}, mx - i)$,否则$r_i = 1$
- 暴力增大r_i
- 更新mx, p

Manacher

- 对于一个i,若$mx > i$,则$r_i = \min(r_{2p-i}, mx - i)$,否则$r_i = 1$
- 暴力增大r_i
- 更新mx, p
- 如何处理偶回文串？No comment

回文计数1

- 给定一个长度为n的字符串S,求前k长的长度为奇数的回文子串长度之积,对一个正儿八经的质数取模
- $n \leq 10^7, k \leq n^2$

回文计数1

- 给定一个长度为n的字符串S,求前k长的长度为奇数的回文子串长度之积,对一个正儿八经的质数取模
- $n \leq 10^7, k \leq n^2$
- 当然先跑一遍Manacher

回文计数1

- 给定一个长度为n的字符串S,求前k长的长度为奇数的回文子串长度之积,对一个正儿八经的质数取模
- $n \leq 10^7, k \leq n^2$
- 当然先跑一遍Manacher
- 我们考虑求出来每种长度的回文子串的个数,记为cnt_i

回文计数1

- 给定一个长度为n的字符串S,求前k长的长度为奇数的回文子串长度之积,对一个正儿八经的质数取模
- $n \leq 10^7, k \leq n^2$
- 当然先跑一遍Manacher
- 我们考虑求出来每种长度的回文子串的个数,记为cnt_i
- 我们每次可以确定中心为i的回文串的长度区间

回文计数1

- 给定一个长度为n的字符串S,求前k长的长度为奇数的回文子串长度之积,对一个正儿八经的质数取模
- $n \leq 10^7, k \leq n^2$
- 当然先跑一遍Manacher
- 我们考虑求出来每种长度的回文子串的个数,记为cnt_i
- 我们每次可以确定中心为i的回文串的长度区间
- 差分+前缀和即可

APIO2014 Palindrome

- 给你一个由小写拉丁字母组成的长度为n的字符串 S。我们定义 S 的一个子串的存在值为这个子串在 S 中出现的次数乘以这个子串的长度。
- 对于给你的这个字符串 S，求所有回文子串中的最大存在值。
- $n \leq 3 * 10^5$

APIO2014 Palindrome

- 给你一个由小写拉丁字母组成的长度为n的字符串 S。我们定义 S 的一个子串的存在值为这个子串在 S 中出现的次数乘以这个子串的长度。
- 对于给你的这个字符串 S，求所有回文子串中的最大存在值。
- $n \leq 3 * 10^5$
- 这个就是一个...回文树(即回文自动机)啦...上次讲过啦

Palindromic Automation

- 回文树有两个根，分别是odd和even
- odd的长度是0,even的长度是1

Palindromic Automation

- 回文树有两个根，分别是odd和even
- odd的长度是0,even的长度是1
- 每个点代表一个回文子串，此时$i \to ch_c$的一步转移代表的是在s_i左右各加上一个字符c，组成新的回文子串

Palindromic Automation

- 回文树有两个根，分别是odd和even
- odd的长度是0,even的长度是1
- 每个点代表一个回文子串，此时$i \to ch_c$的一步转移代表的是在s_i左右各加上一个字符c，组成新的回文子串
- 回忆AC自动机的fail指针
- 我们给回文树定义一个fail指针，指向该点最长回文后缀对应所在的点

Palindromic Automation

- 特别地，odd的fail指向自己,even的fail指向odd

Palindromic Automation

- 特别地，odd的fail指向自己,even的fail指向odd
- 我们考虑如何构建这个回文树
- 设插入的新字符为S_i,我们暴力顺着fail跳找到一个$S_{i-fail_x-1} = S_i$的x，把新字符插在x的孩子

Palindromic Automation

- 特别地，odd的fail指向自己,even的fail指向odd
- 我们考虑如何构建这个回文树
- 设插入的新字符为S_i,我们暴力顺着fail跳找到一个$S_{i-fail_x-1} = S_i$的x，把新字符插在x的孩子
- 如何定向他的fail指针？再跳一步即可

Palindromic Automation

- 特别地，odd的fail指向自己,even的fail指向odd
- 我们考虑如何构建这个回文树
- 设插入的新字符为S_i,我们暴力顺着fail跳找到一个$S_{i-fail_x-1} = S_i$的x，把新字符插在x的孩子
- 如何定向他的fail指针？再跳一步即可
- 复杂度为什么是对的？

Palindromic Automation

- 对于APIO2014的这道题,我们只需要将x的cnt累加到$fail_x$即可

2016 Multi-University Training Contest 5:Interesting

- 给定一个长度为n字符串S,求出所有的$i < j < k, S_{i...j}, S_{j+1...k}$都是回文串的$(i, j, k)$的$i * k$之和
- 5组数据,$n \leq 10^6$

2016 Multi-University Training Contest 5:Interesting

- 给定一个长度为n字符串S,求出所有的$i < j < k, S_{i...j}, S_{j+1...k}$都是回文串的$(i, j, k)$的$i * k$之和
- 5组数据,$n \leq 10^6$
- 当然这个题可以通过Manacher和一番精彩操作来实现,这里不细述

2016 Multi-University Training Contest 5:Interesting

2016 Multi-University Training Contest 5:Interesting

- 注意到上面提到的这种数据结构可以很好的实现求出以i结尾的回文串的相关信息，枚举j即可

2016 Multi-University Training Contest 5:Interesting

- 注意到上面提到的这种数据结构可以很好的实现求出以i结尾的回文串的相关信息，枚举j即可
- 设左右各有长度为a和b回文串，
  $\sum i * k = \sum\sum\sum(j-a)*(j+b)$
- 设左端长度和为ls,右端为rs。左端个数和为ln,右端个数和为rn

2016 Multi-University Training Contest 5:Interesting

- 注意到上面提到的这种数据结构可以很好的实现求出以i结尾的回文串的相关信息，枚举j即可
- 设左右各有长度为a和b回文串，
  $\sum i * k = \sum\sum\sum (j-a)*(j+b)$
- 设左端长度和为ls,右端为rs。左端个数和为ln,右端个数和为rn
- $\sum\sum\sum (j-a)*(j+b) =$
  $\sum_{j=1}^{n} ps*ls*j^2 + j*(rn*ln - rs*lb - rn - ls) - rn*(ls-ln)$