知识精炼（二）

主讲人：邓哲也

BZOJ 4755 扭动的回文串

有两个长度均为 N 的字符串 A 和 B。

扭动的字符串 S(i,j,k) 定义为 A[i..j]+B[j..k]

若 A=’xyz’，B=‘uvw’，则 S(1,2,3)=‘xyvw’

定义扭动的回文串为如下情况中的一个；

A 中的一个回文串

B 中的一个回文串

某一个回文的扭动字符串S(i,j,k)

求最长的扭动回文串

1≤N≤100000，字符串只含大写字母

BZOJ 4755 扭动的回文串

样例：

5

ABCDE

BAECB

答案：

5

最长的扭动回文串是 S(2,3,5) = BCECB

BZOJ 4755 扭动的回文串

对于前两种情况，我们只需要用 manacher 算法求出每个位置的回文扩展半径，然后取一个最大值即可。

对于第三种情况，需要在两个字符串中各取一段拼起来。

我们可以把答案字符串分解为 STS’

其中 S’ 是 S 的反串，T 是回文串。

BZOJ 4755 扭动的回文串

可以发现一定是 S 在一个串，TS’ 在另一个串；或者 ST 在一个串，S’在另一个串。

且 T 一定是极大的回文子串。

思考：为什么？

BZOJ 4755 扭动的回文串

那么我们只要枚举极大回文子串 T 即可。

假设中心在 A 中，枚举中心 i。

假设A[i-p[i], i+p[i]] 是极大回文子串。

1. 枚举 A[1..i-p[i]-1] 的后缀 和 B[i + p[i]..n] 的前缀最长能匹配上多少。
2. 枚举 A[i+p[i]+1..n] 的前缀和 B[1..i-p[i]] 的后缀最长能匹配上多少。

这两部分的最大值 * 2 + 极大回文子串的长度就是答案了。

BZOJ 4755 扭动的回文串

枚举两个字符串能匹配上几位。

BKDRHash！

预处理时间复杂度 O(n)

得到一个子串的 Hash 值的时间复杂度是 O(1) 的。

故可以采用二分+ Hash 的方法求这一部分的答案。

总的时间复杂度是 O(n log n)。

BZOJ 2084 Antisymmetry

给定一个长度为 n 的01串，问有多少个子串满足翻转并取反后和原来一样。

比如0101翻转并取反后还是和原来一样。

BZOJ 2084 Antisymmetry

只要定义 0 = 1， 0 ≠ 0，1 ≠ 1 即可。

跑一遍 Manacher 即可。

以 0/1 扩展的 p[i] 一定是 0 （即没有奇数长度的合法子串）

以 # 扩展的 p[i] 累加起来即可。

BZOJ 2084 Antisymmetry

只要定义 0 = 1， 0 ≠ 0，1 ≠ 1 即可。

跑一遍 Manacher 即可。

以 0/1 扩展的 p[i] 一定是 0 （即没有奇数长度的合法子串）

以 # 扩展的 p[i] 累加起来即可。

BZOJ 3790 神奇项链

母亲节就要到了，小 H 准备送给她一个特殊的项链。这个项链可以看作一个用小写字母组成的字符串，每个小写字母表示一种颜色。为了制作这个项链，小 H 购买了两个机器。第一个机器可以生成所有形式的回文串，第二个机器可以把两个回文串连接起来，而且第二个机器还有一个特殊的性质：假如一个字符串的后缀和一个字符串的前缀是完全相同的，那么可以将这个重复部分重叠。例如：aba和aca连接起来，可以生成串abaaca或 abaca。现在给出目标项链的样式，询问你需要使用第二个机器多少次才能生成这个特殊的项链。

样例：

abcdcba　　（答案：0）

abacada　　（答案：2）

abcdef　　（答案：5）

BZOJ 3790 神奇项链

题目的意思就是用尽可能少的回文子串去覆盖原串。

我们可以先跑一遍 Manacher 算法得到每个位置的覆盖半径。

问题转化为了有许多个区间 [i-p[i], i+p[i]]

要选出尽可能少的区间来覆盖 [1, n] 区间。

BZOJ 3790 神奇项链

贪心即可。

对所有的区间按照左端点从小到大排序。

每次都选右端点最远的那个区间。

设当前覆盖到的右端点为 right。

每次找出左端点在[1, right] 中的右端点的最大的那个线段，再做一次覆盖。

时间复杂度 O(n log n)

下节课再见