$O(n)$ 预处理， $O(1)$ 得到每个子串 Hash 值

$pre[i]$: $S[1 \dots i]$ 的 Hash 值

$$pre[i+1] = pre[i] \times B + (S[i] - '0')$$

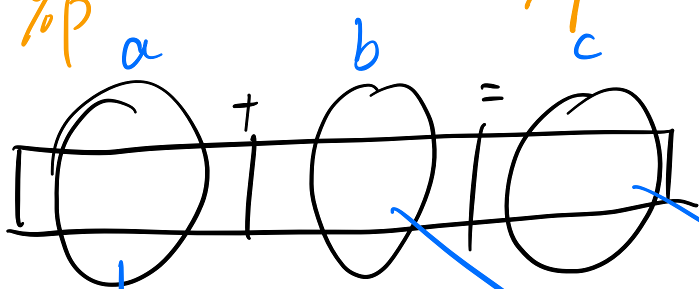
$pwB[i] = B^i \% p$ $O(n)$ 预处理

$S[l \dots r]$ 的哈希值：

$$\underbrace{pre[r]}_{\%p} - \underbrace{pre[l-1] \times B^{r-l+1}}_{\%p}$$

$pwB[r-l+1]$

$B=10$



hash 值： $(a\%p + b\%p)\%p = c\%p$ （判断）

双模数 取两个不同 p
分别判断一次.

暴力: $O(n^2)$
枚举 "+" 和 "=" 位置, 再 $O(1)$ 判断.

正解 $O(n)$

简化: 假设没有 0

如果 $a>b$，$a+b=c$，a 是 k 位数

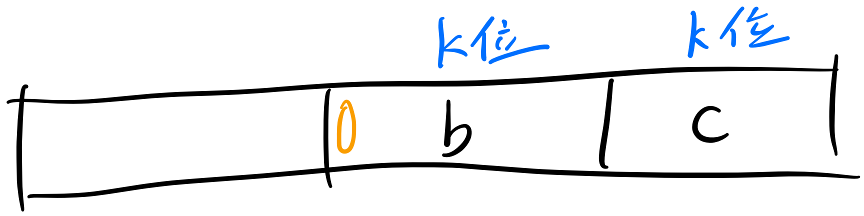
c 可能是几位数？ $\begin{cases} k \\ k+1 \end{cases}$

对于 $a<b$，类似。

1. 枚举"="位置（k 从号到 n）

1) $a>b$，"+"可能在 k 或 $k-1$ 处

2) $a<b$，"+"可能在 $n-2k$ 或 $n-2k+1$ 处



特判！

过程中特判 a、b、c 是否有前导 0.

P3501　　输入 S，取反再翻转得 T

枚举子串 S[l...r]

l = 1 to n　　　$O(n^2)$

r = l to n

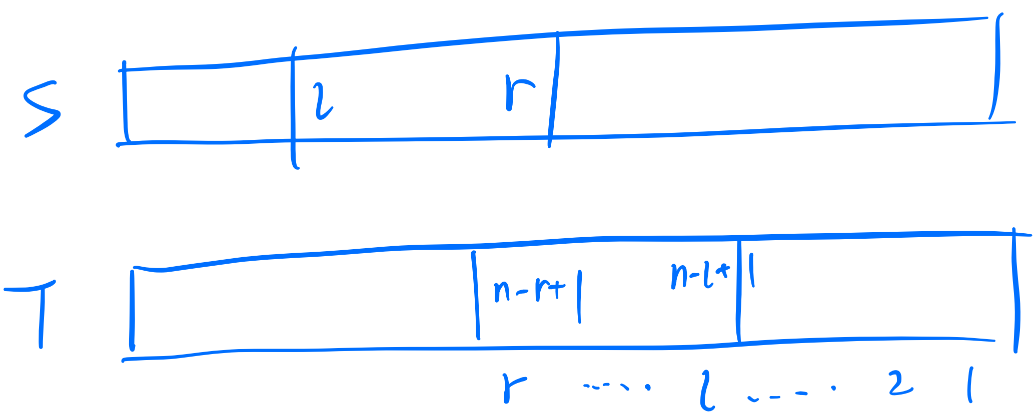
O(1) 判断是不是"反对称"串

S[l...r] 是"反对称"

当且仅当　S[l...r] = T[n-r+1 ... n-l+1]

求 Hash

O(1) 判断.



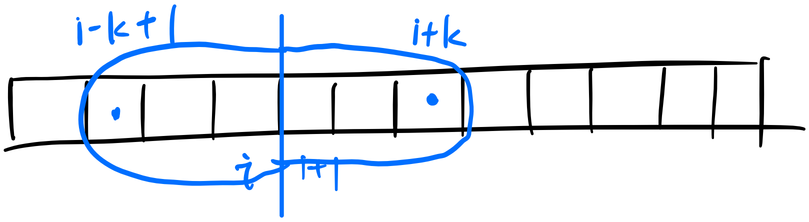
正解.

XXX0XXX

YYY1YYY

1. "反对称"串长度一定是偶数

枚举"反对称"串的中轴线.



枚举"反对称"串一半的长度 k

O(1) 判断 s[i-k+1 ... i+k] 是不是"反对称"

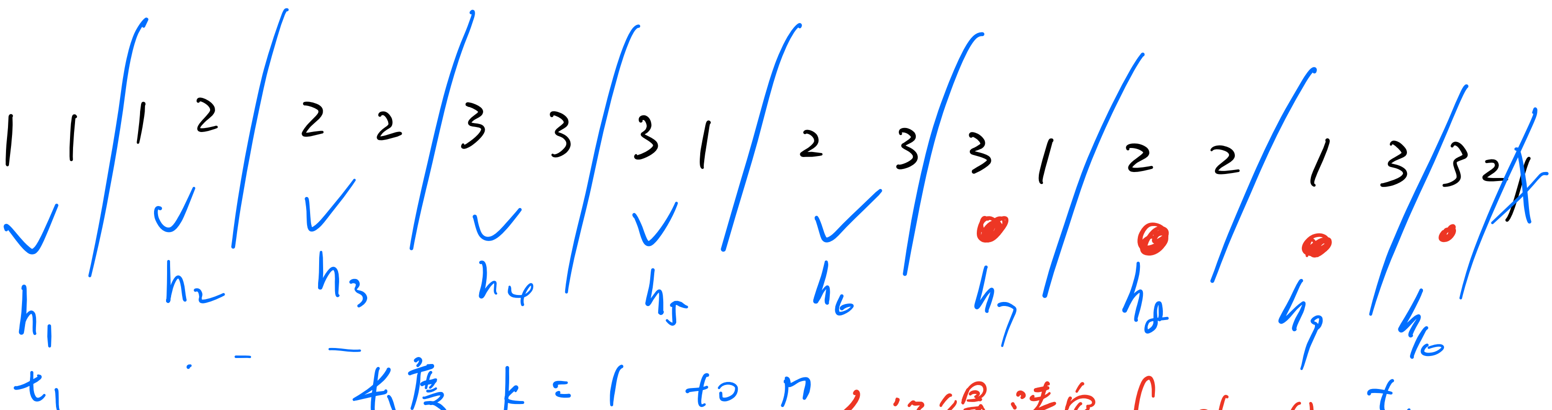
(例) 11

0 1 0 1 0 1

1 0 1 0

0 1

k 可以二分.



长度 k=1 to n < 记得清空 f.clear()

位置 i=1, i≤n, i+=k

O(1) 判断, 用哈希.

$$n+\frac{n}{2}+\frac{n}{3}+\cdots+\frac{n}{n}\le n+n\ln n$$

$$O(n\ln n)$$

unordered_map<long long, bool> f

每算出一个 t_i 和 h_i，判断

$f.count(t_i) \,||\, f.count(h_i)$ 是否为真，

若是，说明出现过

若否，说明没出现，令 $f[h_i] = true$

（别给 $f[t_i]$ 赋值）

最后，$f.size()$ 就是有几种不同的