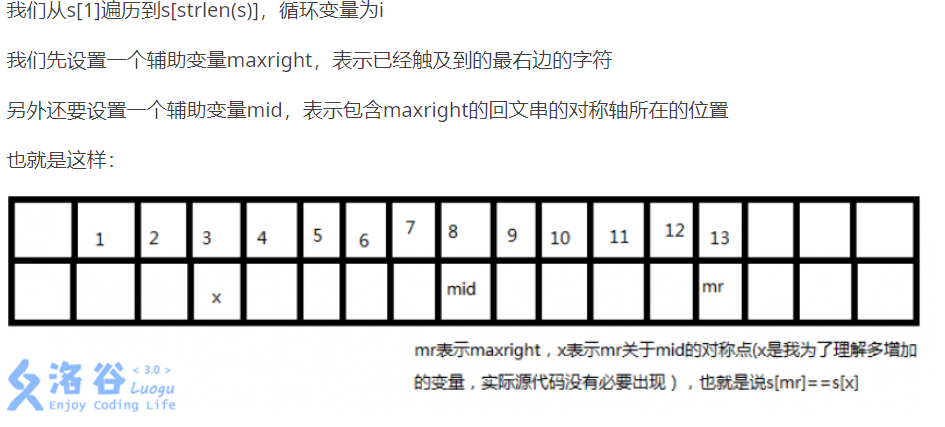
manacher算法luogu3805

给出一个只由小写英文字符组成的字符串S，求S中最长回文串的长度。（n<=10^7）

首先，对于一个abccba的字符串，我们可以从他们的回文串的回文中心入手，记录每个回文中心能够扩展到的最大半径，

记录为p[i]，但是像上面这种字符串，中间是个空格，所以我们将它化为 ~#A#B#C#C#B#A#这种形式，使得其中间可以记录



1.当i在maxright左边且在mid右边时：设i关于mid的对称点为j，

当p[j]比较小，没有能够扩展超过x，那么显然p[i]就等于p[j],j扩展超过x时，由于maxright右边不和x左边相同，所以p[i]最大为r-i+1，然后再尝试扩产p[i]

2. 当i在maxright右边，自己重新扩产p[i]就好，又有新的maxright和mid

代码如下：

#include<bits/stdc++.h>

#define ll long long

#define inf 1e18

#define mn 11000005

using namespace std;

ll ans=0,n,m,p[2\*mn];

char a[mn],s[mn\*2];

void change()

{ll i;

s[0]='~';

s[1]='#';

for(i=0;i<n;i++)

{

s[i\*2+2]=a[i];

s[i\*2+3]='#';

}

n=n\*2+1;

}

void manacher()

{ll i,r=0,mid;

for(i=1;i<=n;i++)

{

if(i<=r)p[i]=min(p[mid\*2-i],r-i+1);

while(s[i-p[i]]==s[i+p[i]])p[i]++;

if(p[i]+i-1>r){r=p[i]+i-1;mid=i;}

}

}

int main()

{

ll x,y,z,i,j,k;

char ch;

scanf("%s",a);

n=strlen(a);

change();

manacher();

for(i=1;i<=n;i++)

ans=max(ans,p[i]);

printf("%lld",ans-1);

return 0;

}