

马拉车模板

```
#include<bits/stdc++.h>
using namespace std;
typedef long long LL;
const int maxn=1000010;
char str[maxn];
char s[maxn*2];
int p[maxn*2];
int len;
void Manacher()
{
    int i;
    s[0]='$';
    s[1]='#';
    for(i=0;i<len; i++) {
        s[i*2+2]=str[i];
        s[i*2+3]='#';
    }
    len=i*2+2;
    s[len]='\0';
    int MaxL,id=0;
    MaxL=0;
    memset(p,0,sizeof(p));
    for(int i=0; i<len; i++) { //id是延申到最右边回文串的中心位置
        if(p[id]+id>i) //i在最大回文串内
            p[i]=min(p[2*id-i],p[id]+id-i); //串内对应位置的回文半径，不能超出回文串，超出部分一点一点找
        else
            p[i]=1; //在外面就设一，老老实实一点一点找
        while(s[i+p[i]]==s[i-p[i]]) //往外找
            p[i]++;
        if(p[i]+i>p[id]+id) { //比较记录延申最右边回文串的中心位置
            id=i;
        }
        if(p[i]-1>MaxL) //p[i]是s[i]（带#号）为中心，最大回文半径长度，p[i]-1则恰好是对应原串（无#号）该位置为中心最大回文串的长度，可举例
            MaxL=p[i]-1;
    }
}
```