科大讯飞笔试编程第三题

小红定义一个字符串是“好串”，当且仅当该字符串的长度不小于2，且首尾相同。  
  
一个仅包含小写字母的字符串，长度不超过200000。  
  
输出描述  
  
如果无法切割且该字符串本身不是好串，请输出-1。否则输出最终的好串数量。  
  
思路：正常dp转移是dp[i]=max(dp[i],dp[x]+1), a[x]=a[i], 复杂度是O(n^2)的  
  
改进：dp[i]表示[1,i]中最大分割次数；trans[i-1]堆维护第i个字母后面一个位置（假设是x）dp[x]的最大值；dp[i]的转移位置由trans[a[i]]来维护，转移的过程中必须得保证转移的位置合法。（今天才发现不用堆也行，只要最大值就行）  
  
这题确实有点东西，第一次感受到笔试编程题带来的压迫感  
  
#include  
using namespace std;  
const int maxn=2e5+100;  
priority\_queue trans[26];//以a[i]后面一格的最大dp值  
string s;  
int a[maxn];  
int dp[maxn],n;   
int main(){  
cin>>s;  
    for(int i=1;i<=s.size();i++){  
        a[i]=s[i-1]-'a';  
    }  
    n=s.size();  
  
    memset(dp,-1,sizeof(dp));  
    dp[0]=0;  
    for(int i=1;i<=n;i++){  
        if(trans[a[i]].empty()){  
        if(dp[i-1]!=-1){//身后元素是合法状态   
        trans[a[i]].push(dp[i-1]);   
}  
continue;  
}  
        int mx=trans[a[i]].top();  
        dp[i]=mx+1;  
        if(dp[i-1]!=-1){//身后元素是合法状态   
    trans[a[i]].push(dp[i-1]);   
}  
    }  
    cout<