首先我们只能从左侧删除字符串，那么我们剩下的只能是字符串右侧的部分。

而最后两个字符串剩下的部分长度是相等的，并且每一位上的字母都是匹配的，所以实际上就是两个字符串从右往左匹配的部分。

我们所求的是最少删除次数，那么对应的也就是从右往左的最长匹配部分。

用两个字符串的总长度减去最长匹配部分长度\*2即为答案。

C语言标程:

#include<stdio.h>

#include<string.h>

**const** **int** maxn=2e5+7;

**char** a[maxn],b[maxn];

**int** main()

{

    scanf("%s%s",a,b);

**int** lena=strlen(a),lenb=strlen(b);

**int** temp=1;//temp表示从倒数第几位开始两个字符串不再匹配

**while**(lena-temp>=0&&lenb-temp>=0)

    {

**if**(a[lena-temp]==b[lenb-temp]) temp++;

**else** **break**;

    }

    printf("%d\n",lena+lenb-2\*(temp-1));

}

C++标程:

#include<bits/stdc++.h>

#define IOS ios::sync\_with\_stdio(0); cin.tie(0); cout.tie(0);

//取消cin,cout同步，可以使得cin,cout提速。

//取消同步后在cf上与scanf,printf同速度，其他大部分oj上仍然慢与scanf,printf

**using** **namespace** std;

**int** main()

{

    IOS

    string a,b;cin>>a>>b;

**int** lena=a.size(),lenb=b.size();

**int** temp=1;//temp表示从倒数第几位开始两个字符串不再匹配

**while**(lena-temp>=0&&lenb-temp>=0)

    {

**if**(a[lena-temp]==b[lenb-temp]) temp++;

**else** **break**;

    }

    cout<<lena+lenb-2\*(temp-1)<<endl;

}