# 题目

给你一个仅由大写英文字母组成的字符串，你可以将任意位置上的字符替换成另外的字符，总共可最多替换 k 次。在执行上述操作后，找到包含重复字母的最长子串的长度。

**注意:**

字符串长度 和 k 不会超过 104。

**示例 1:**

输入:

s = "ABAB", k = 2

输出:

4

解释:

用两个'A'替换为两个'B',反之亦然。

**示例 2:**

输入:

s = "AABABBA", k = 1

输出:

4

解释:

将中间的一个'A'替换为'B',字符串变为 "AABBBBA"。

子串 "BBBB" 有最长重复字母, 答案为 4。

# 分析

## 方法一：滑动窗口

class Solution {

public:

int characterReplacement(string s, int k) {

int count[26] = {0};//字符->字符数量的映射

int left=0,right=0,ret=0,maxCount=0;

while(right<s.size())

{

count[s[right]-'A']++;//统计某个字母出现次数

maxCount = max(maxCount,count[s[right]-'A']);//当前窗口可以容纳的最多字符数量

if(right-left+1-maxCount>k)//对于替换字符个数大于k的情况需要缩小窗口(滑动窗口大小-出现次数最多的字符的个数=可替换的字符个数,即k值)

{

count[s[left]-'A']--;

left++;

}

ret = max(ret,right-left+1);

right++;

}

return ret;

}

};