预处理

交替枚举

遇到的第一个codeforces需要带输入输出文件的题目

#include <cstdio>

#include <cstring>

#include <iostream>

using namespace std;

typedef long long ll;

char a[405][405];//原表

int num[405][405];//前缀和，a[i][j]表示第i行，1 ~ j列字符a的数量

int q[30];//计算当前子表每行中最左列等于最右列的数量

int main()

{

freopen("input.txt", "r", stdin);

freopen("output.txt", "w", stdout);

int n,m,kk;

scanf("%d%d%d",&n,&m,&kk);

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

scanf("%s",a[i]+1);

memset(num,0,sizeof(num));

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

for(int j=1; j<=m; j++)

{

num[i][j]=num[i-1][j]+num[i][j-1]-num[i-1][j-1];//注意是-num[i-1][j-1]

num[i][j]+=(a[i][j]=='a'?1:0);

//num[i][j]代表 宽为i 长为j的矩阵中a的数量

//预处理一下二维的前缀合

}

ll ans=0;

int q[30];//选取矩阵中各个字母的出现的对数

for(int i=1; i<=n; i++)//矩形上边

for(int j=i+1; j<=n; j++)//矩形的下边

{

int p=1;//p代表矩阵右边

memset(q,0,sizeof(q));

//如果q[2]=4 代表b这个字母在以ij为上下边界

//kp为左右边界(kp在枚举着变动，ij是不变的)的一段中出现过4次

//肯定a[i][k]和a[j][k]为相同字母 a[i][p]和a[j][p]为相同字母，而这个字母出现过4次也就是说ik.jk这条边可以和这里面4个点组成矩形，矩形数量就加4

for(int k=1; k<=m; k++) //矩阵左边

{

if(a[i][k]!=a[j][k]) continue;//不满足矩形四个点相同

q[a[i][k]-'a']--;//算能和几个点组成矩形的时候不能算它本身，所以减1

while(p<=m && num[j][p]-num[i-1][p]-num[j][k-1]+num[i-1][k-1]<=kk)

{

if(a[i][p]==a[j][p])

q[a[i][p]-'a']++;

p++;//矩形满足a个数小于k的要求，矩阵右边增大，直到不满足，那么增大左边k++

}

if(q[a[i][k]-'a']>0) ans+=q[a[i][k]-'a'];//因为矩形左边固定，

}

}

printf("%I64d\n",ans);

return 0;

}