

P3392 涂国旗

题目描述

某国法律规定，只要一个由 $N \times M$ 个小方块组成的旗帜符合如下规则，就是合法的国旗。

从最上方若干行（至少一行）的格子全部是白色的；

接下来若干行（至少一行）的格子全部是蓝色的；

剩下的行（至少一行）全部是红色的；

现有一个棋盘状的布，分成了 N 行 M 列的格子，每个格子是白色蓝色红色之一，小 a 希望把这个布改成该国国旗，方法是在一些格子上涂颜料，盖住之前的颜色。

小 a 很懒，希望涂最少的格子，使这块布成为一个合法的国旗

输入格式

第一行是两个整数 N, M 。

接下来 N 行是一个矩阵，矩阵的每一个小方块是 W（白），B（蓝），R（红）中的一个

输出格式

一个整数，表示至少需要涂多少块

输入样例

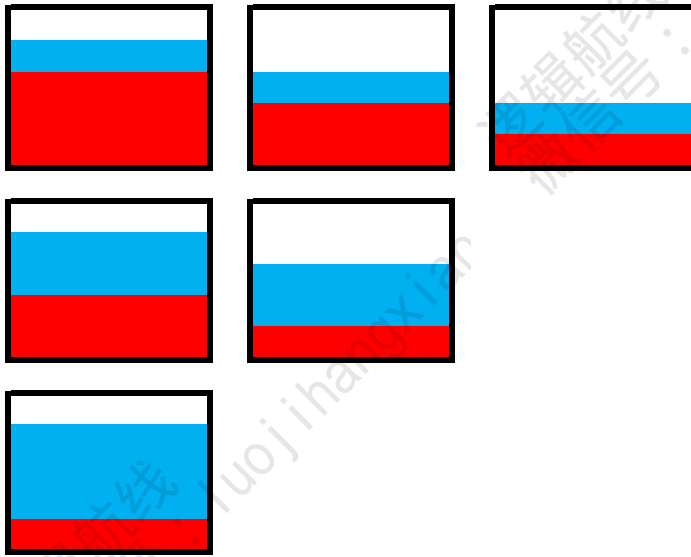
```
4 5
WRWRW
BWRWB
WRWRW
RWBWR
```

输出样例

```
11
```

解析

整个旗子分成三个颜色，上为白，中为蓝，下为红。如下所示：



通过观察上面几幅图，我们发现需要涂改的块数与三种颜色的限定区域有关，因此，我们只需要不断的枚举白色和蓝色的限定范围（限定了两种后，红色会自动限定），然后统计所需要涂抹的块数即可。

编码

```
#include <bits/stdc++.h>

int n, m, ans = 1e9;
char a[52][52];

int main() {
    scanf("%d%d", &n, &m);
    // 整行读入国旗信息
    for (int i = 0; i < n; i++)
        scanf("%s", a[i]);

    // 不断的尝试白色区域的下边界，注意至少要预留两行
    for (int i = 0; i < n - 2; i++) {
        // 不断尝试蓝色区域的下边界，注意至少要预留一行
        for (int j = i + 1; j < n - 1; j++) {
            // 重置数据
            int tot = 0;
            // 计算染白需要几步
            for (int x = 0; x <= i; x++) {
                for (int y = 0; y < m; y++) {
                    if (a[x][y] != 'W') {
                        tot++;
                    }
                }
            }
            // 计算染蓝需几步
            for (int x = i + 1; x <= j; x++) {
                for (int y = 0; y < m; y++) {
```

```

        if (a[x][y] != 'B') {
            tot++;
        }
    }
}
//计算染红需几步
for (int x = j + 1; x < n; x++) {
    for (int y = 0; y < m; y++) {
        if (a[x][y] != 'R') {
            tot++;
        }
    }
}
//更新最小步数
if (tot < ans) {
    ans = tot;
}
}
}
printf("%d", ans);
return 0;
}

```

逻辑航线培优教育，信息学奥赛培训专家。

扫码添加作者获取更多内容。

