

# 백준 1018 체스판 다시 칠하기

202020988 조아영

# 문제

$N \times M$ 크기의 보드 : B, W으로 칠해짐

->  $8 \times 8$ 크기의 체스판 만들기

(변을 공유하는 두 사각형을 다른색으로 칠해져 있어야 함)

->  $8 \times 8$ 크기로 잘라낸 후, 다시 칠하기

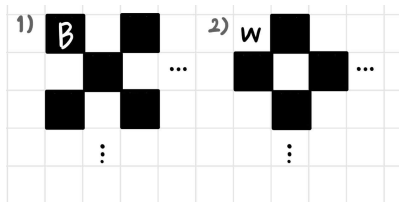
-> 다시 칠해야하는 정사각형의 개수 구하기

입력 : N, M, 보드 색의 상태

출력 : 다시 칠해야 하는 정사각형의 최소 개수

# 아이디어

“체스판을 색칠하는 경우는 두 가지뿐이다. 하나는 맨 왼쪽 위 칸이 흰색인 경우, 하나는 검은색인 경우이다.”



- > 맨 왼쪽 위칸의 색을 기준으로 두 경우의 체스판으로 나눔
- > 맞지 않는 색의 개수를 셈

# 아이디어

-> 문제에서 주어진 예제는 맞았으나, 제출 해본 결과 틀림

이전, 아이디어의 문제

: 오히려, 맨 왼쪽 위칸의 색을 바꾸어 다시 칠하는 경우가 더 최소한의 색을 칠하는 방법인 경우도 존재

-> 두 경우를 모두 세서, 최소값을 저장

# 코드

- 헤더파일 선언
- N,M / 보드판 / 여러변수들 선언
- N,M 입력
- 행의 수 N 만큼 반복 보드판의 상태( 문자열 ) 입력
- 출력 결과인 min을 초기화  
(8\*8=64보다 큰 65로)

```
#include <stdio.h>

int main()
{
    int M, N;
    char board[50][51];
    int cnt1, cnt2, flag, min, temp;

    scanf("%d%d", &N, &M);

    for (int i = 0; i < N; i++)
        scanf("%s", board[i]);

    min = 65;
```

```
for (int i = 0; i <= N - 8; i++)
{
```

→ i, j의 값 변화 시켜가면서, 체스판 8\*8로 자르기

```
    for (int j = 0; j <= M - 8; j++)
    {
```

```
        cnt1 = cnt2 = 0;
    }
    cnt1 = cnt2 = 0; → 변수 초기화
```

```
    for (int l = i; l < i+8; l++) {
```

→ l, k의 값 변화 시켜가면서, 체스판 8\*8 모든 칸 탐색

```
        for (int k = j; k < j+8; k++)
        {
```

```
            1) if ((board[l][k] == 'B' && ((l - i) + (k - j)) % 2 != 0)
                || (board[l][k] == 'W' && ((l - i) + (k - j)) % 2 == 0))
                cnt1++;
```

```
            2) if ((board[l][k] == 'B' && ((l - i) + (k - j)) % 2 == 0)
                || (board[l][k] == 'W' && ((l - i) + (k - j)) % 2 != 0))
                cnt2++;
        }
```

→ 1), 2) 각각의 체스판과  
맞지 않는 색의 개수 셈

```
    }
```

```
    if (cnt1 >= cnt2)
```

```
        temp = cnt2;
```

```
    else
```

```
        temp = cnt1;
```

→ cnt1, cnt2 중 작은 값 temp에 저장

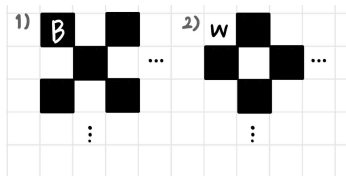
```
    if (min > temp)
```

```
        min = temp;
```

→ 최소값 저장

```
}
```

```
}
```



결과 출력 → `printf("%d", min);`

감사합니다