

RGB 거리

<https://www.acmicpc.net/problem/1149>

- $dp[i][j]$ = i 번 집을 j 색으로 칠하였을 때, $1 \sim i$ 번 집을 칠하는데 필요한 최소 비용
- R : $j = 0$ / G : $j = 1$ / B : $j = 2$
- R : $dp[i][0] = \min(dp[i-1][1], dp[i-1][2]) + A[i][0]$
- G : $dp[i][1] = \min(dp[i-1][0], dp[i-1][2]) + A[i][1]$
- B : $dp[i][2] = \min(dp[i-1][0], dp[i-1][1]) + A[i][2]$
- 초깃값 : $i == 0$ 일때 비용은 0

```
int solve(int i, int j) {
    if (i == 0) return 0;

    if (dp[i][j] != -1) return dp[i][j];

    dp[i][j] = 1e9;
    if(j==0) dp[i][j] = min(solve(i - 1, 1), solve(i - 1, 2)) + A[i][0];
    if(j==1) dp[i][j] = min(solve(i - 1, 0), solve(i - 1, 2)) + A[i][1];
    if(j==2) dp[i][j] = min(solve(i - 1, 0), solve(i - 1, 1)) + A[i][2];

    return dp[i][j];
}
```



RGB 거리

<https://www.acmicpc.net/problem/1149>

```
for (int i = 1; i <= N; ++i) {  
    dp[i][0] = min(dp[i - 1][1], dp[i - 1][2]) + A[i][0];  
    dp[i][1] = min(dp[i - 1][0], dp[i - 1][2]) + A[i][1];  
    dp[i][2] = min(dp[i - 1][0], dp[i - 1][1]) + A[i][2];  
}
```

