

- 문제: C와 같이 n과 S가 주어진다.

이번에는 연속되는 문자 나오지 않게 새로운 S를 만들어내는 문제.
(최적 cost).

- 접근: DP인가.. 완탐인가 고민하다가

그리디 느낌으로 앞뒤와 다른 문자로 바꿔주는 식으로 짰더니 AC.
(문자가 총 3개니까 앞뒤와 다른 가운데 문자가 무조건 있음).

- 구현

```
int n;  
cin >> n;  
string S;  
cin >> S;
```

```
string C = "RGB";
```

```
S += '$'; // n번째까지 검사하게 되므로 (i+1 때문이).
```

```
int ans = 0;
```

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

```
    if (S[i-1] == S[i]) {
```

```
        ++ans;
```

```
        int p = 0;
```

```
        while (C[p] == S[i-1] || C[p] == S[i+1]) ++p;
```

```
        S[i] = C[p];
```

```
    }
```

↳ 앞, 뒤와 다른 모델 C의 문자 인덱스 얻기 위함.
(가운데로 0~2 찾을 수도 있으나 이게 더 간단함.)

```
S.pop-back(); // 문자열도 벡터 구조라서 pop-back()으로 $ 빼줌.
```

```
cout << ans << '\n' << S << '\n';
```