

- 문제 : 'R', 'G', 'B'로 이루어진 문자열을
RGB 또는 BGR 순으로 바꾸기 위해 필요한 문자 수와
최적의 새로운 string을 구하는 문제.

- 접근 : 6가지 길이 n만큼 만들어서 비교하고 다른 정도가
가장 작을 때의 cnt 값과 그 문자열을 출력하면 된다.

- 구현 : 결국 ^{n자리 새로운 문자열을 만들기 위한 가장 짧은 모델.} 필요한 모델은 "BGR" ~ "RGB" 6가지.

=> 순열로 생각할 수 있음.

* tip. 문자열도 벡터처럼 순열을 돌릴 수 있음.

↳ push_back도 가능함 물론 char형만.
sort도 가능함 (begin(), end()가 존재).

```
int n; cin >> n;  
string s; cin >> s;  
string t = "BGR"; // 인접치순 정렬된 모델.  
int mn = INT_MAX;  
string ans = "";
```

```
do {  
    int cnt = 0;  
    for (int i = 0; i < n; ++i) {  
        if (t[i % 3] != s[i]) cnt++;  
    }
```

```
    if (cnt < mn) {  
        mn = cnt;  
        for (int i = 0; i < n; ++i)  
            ans += t[i % 3];  
    }
```

```
} while (next_permutation(t.begin(), t.end()));  
cout << mn << '\n' << ans << '\n';
```