

# 올바른 괄호 문자열

<https://www.acmicpc.net/problem/3012>

- $dp[i][j]$  =  $i \sim j$  까지 문자열을 이용해서 만들 수 있는 올바른 괄호 문자열의 개수
- $i$ 번째에 있는 왼쪽 괄호와 짝이 맞는 오른쪽 괄호의 위치를  $k$ 라고 하였을 때  $(i+1, k-1)$ ,  $(k+1, j)$ 로 문제를 나눌 수 있다.
- $dp[i][j] = dp[i+1][k] * dp[k+1][j]$



- $i$ 가 왼쪽 괄호,  $k$ 가 오른쪽 괄호인 경우  $(i+1, k-1)$  문제로 분할 할 수 있다.  $(k+1, j)$ 범위 또한 탐색 해야 한다.

```
//dp[i][j] = i~j 까지 문자열을 이용해서 만들 수 있는 올바른 괄호 문자열의 개수
int solve(int i, int j) {
    if (i > j) {
        return 1;
    }

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

    dp[i][j] = 0;
    for (int k = i + 1; k <= j; k++) {
        for (int l = 0; l < 3; l++) {
            if (A[i] == open[l] || A[i] == '?') {
                if (A[k] == close[l] || A[k] == '?') {
                    int temp = solve(i + 1, k - 1) * solve(k + 1, j);
                    if (dp[i][j] + temp > mod) {
                        over = true;
                    }
                    dp[i][j] += temp;
                    dp[i][j] %= mod;
                }
            }
        }
    }
    return dp[i][j];
}
```