

3주차 - 2133: 타일 채우기 풀이

202055524 김의현

2025-4-4

1 문제 설명

$3 \times N$ 크기의 벽을 $2 \times 1, 1 \times 2$ 의 타일로 채우는 경우의 수를 구하라.

2 접근법

C 를 경우의 수를 구하는 함수라고 하자.

당연히 N 이 홀수인 경우는 채워넣을 수 없다. 따라서 이때 $C(N) = 0$ 이다.

다음으로 N 이 짝수인 경우를 고려해보자. 먼저 $N = 0$ 일 경우 당연히 $C(0) = 0$ 이다.

$N = 2$ 일 경우 각 타일을 $(2, 1)$ 개 사용하는 경우 2가지, $(3, 0)$ 개씩 사용하는 경우 1가지가 가능하므로 $C(N) = 3$ 이다.

$N = 4$ 인 경우를 생각해보면, $N = 2$ 일때에 비해 2칸이 더 생겼으므로 단순히 $C(4) = C(2) * 3$ 이라 생각할 수 있다. 그러나, 그림 1과 같이 $N = 2$ 일때는 불가능했던 방식으로도 붙일 수 있다. 상/하를 반전한 모습으로 붙인 것도 유효하므로, 2가지 경우가 추가로 가능하다. 따라서 $C(4) = C(2) * 3 + 2 = 11$ 이다.

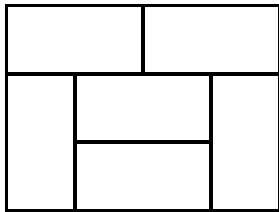


Figure 1: $N = 4$ 일 경우

조금 더 생각해보면, $N \geq 4$ 인 모든 짝수에 대해서 2×1 블록 $\frac{3N}{2} - 2$ 개, 1×2 블록 2개를 사용하면 위와 유사한 패턴으로 붙일 수 있음을 알 수 있다.

따라서, 다음과 같은 점화식을 유도할 수 있을 것이다.

$$C(N) = \left(\sum_{k=2} C(N-2k) * 2 \right) + 3 * C(N-2) \quad (N = 2k, k \in \mathbb{N}, k \geq 2)$$

$$C(N) = 3 \quad (N = 2)$$

$$C(N) = 0 \quad (N = 2k + 1, k \in \mathbb{N})$$

3 풀이

```
#include <iostream>
#include <vector>

int main() {
    std::vector<unsigned> dp(16);
    dp[0] = 1;
    dp[1] = 3;
    for (auto i = 2u; i < 16; ++i) {
        dp[i] += dp[i - 1] * 3;
        for (auto j = 0u; j < i - 1; ++j) {
            dp[i] += dp[j] * 2;
        }
    }

    unsigned n;
    std::cin >> n;
    if (n % 2 == 1 || n == 0) {
        std::cout << 0;
    } else {
        std::cout << dp[n / 2];
    }
}
```