Tiling with Dominoes in C++

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

using namespace std;

int main() {
    int n = 2;

    vector<int> dp(n + 1);
    dp[1] = 1;
    dp[2] = 2;

    for (int i = 3; i <= n; i++) {
        dp[i] = dp[i - 1] + dp[i - 2];
    }

    cout << dp[n] << endl;

    return 0;
}
```

Initial Setup:

- Input: n = 2.
- A dp array of size n+1 is created, i.e., dp[3].

Step 1: Initialize Base Cases

- dp[1] = 1
- dp[2] = 2

At this point, the dp array looks like:

dp = [0, 1, 2]

Step 2: Iterative Calculation

The for loop starts from i = 3 and runs up to n. However, since n = 2, the loop condition i <= n is **not satisfied**. Hence, the loop does **not execute**.

Step 3: Output the Result

The program outputs the value of dp[n], which is dp[2] = 2.

Output:-

```
2
```