

Tiling in C++	
<pre> #include <iostream> using namespace std; int tilingways(int n) { if (n == 0) { return 0; } if (n == 1) { return 1; } return tilingways(n - 1) + tilingways(n - 2); } int main() { cout << tilingways(4) << endl; return 0; } </pre>	<p>Step-by-step Calculation</p> <ol style="list-style-type: none"> 1. tilingways(4): <ul style="list-style-type: none"> o tilingways(3) + tilingways(2) 2. Recursive call: tilingways(3): <ul style="list-style-type: none"> o tilingways(2) + tilingways(1) 3. Recursive call: tilingways(2): <ul style="list-style-type: none"> o tilingways(1) + tilingways(0) 4. Base case reached: tilingways(1) returns 1 (since there is 1 way to tile a 2x1 grid). <ul style="list-style-type: none"> o Base case reached: tilingways(0) returns 0 (no way to tile a 2x0 grid). o Result: tilingways(2) = 1 + 0 = 1 5. Base case reached: tilingways(1) returns 1. <ul style="list-style-type: none"> o Result: tilingways(3) = 1 + 1 = 2 6. Recursive call: tilingways(2): <ul style="list-style-type: none"> o tilingways(1) + tilingways(0) o tilingways(1) returns 1, tilingways(0) returns 0. o Result: tilingways(2) = 1 + 0 = 1 7. Final Calculation: tilingways(4) = 2 + 1 = 3
<p>Output:-</p> <p>3</p>	