Lego (Spoiler)

The point is to split the problem into two parts. The possible configurations of each layer can be generated by brute force and then simply compared with the input data. There are maximally $Q = 2411$ configurations compatible with the input. One should only note that the colors don't increase the complexity; either the color of a block is completely determined by the input or it can have any color (i.e. just multiply by three), so the generation of configurations should be done in "black-and-white".

Without the rule that each block must rest on another, the answer would simply be the product of the numbers of configurations in each layer, but the rule introduces a dependency between the layers. As the limitations for each layer only depend on the closest underlying layer, this problem can be solved with a simple dynamic programming algorithm. The complexity becomes $O(H \cdot Q^2)$ but it is necessary to do the compability test of two configurations in a reasonably efficient way (e.g. by saving the "knob pattern" for each configuration).