INTRO: (🡨 Can you tell me what’s your project is about?)

1. The puzzle has 4 cubes, each face having one of the 4 colors (R, B, W, G). The goal is to put them together into a line so that each long side of the rectangular solid has one and only one occurrence of a color.
2. It’s hard to solve by hand (over 41,000 possible arrangments).
3. We did a bit research, and found a graph theoretical approach to this problem. We used this method to write a problem a computer program that tests for and displays the solution(s) to any set of 4 cubes inputted by user.

DETAILS: (🡨 Ok, tell me more. / Sounds cool. So how does it work?)

[**Basic terminology: vertex, edge, loop, degree**]

To visualize the puzzle more easily, we flatten the 6 faces of each cube into a cross. **<point at the crosses>**

*We’ll use four vertices to represent four colors.*

* 1st observation: If we pick one face to be the top, it strictly follows that its opposite face is gonna be the bottom. Meanwhile, the remaining 4 faces can be freely rotated.

🡪 This is one constraint relationship that is needed for finding solution. In other words, we’ll now represent the faces in terms of “pair of opposite faces.” *If we draw a graph, we’ll draw an edge for each such pair (, connecting the corresponding colors). If two opposite faces have the same color, we’ll draw a loop at that vertex.*

* 2nd observation: Since only the long sides of the rectangular solid matter, the left and right faces (of each cube) do not count towards the solution.

🡪 Our representation of the solution state only needs 2 pairs (out of 3) from each cube. *Since pairs are represented in edges, the edges have to be distinct since one edge can’t be both top-bottom and front-back.*

So, here are the steps:

1. Transform the crosses into a graph that has 3 edges from each cube. We call this a “master graph.”
2. Extract acceptable subgraphs. An acceptable subgraph consists of one edge from each cube, and each vertex has degree-2 (i.e. either 1 loop or 2 edges touch it).
3. Combine any 2 subgraphs that are edge-distinct (if there are any). (Explain why edge-distinct). Each of these combinations is a solution.
4. Map the combined subgraphs to the physical solution. Pay attention to orientation.

Those are the theoretic steps, and here’s our program. Do you want to try?