

General Description of the design of my app:

My app tries to provide an easy-to-understand tutorial to those beginners who haven't solved any Rubik's cubes in their life ever. My tutorial solves the cube layer by layer and splits the whole process into seven steps. Before each step, the goal of the step is demonstrated so that the users can know what they should do during each step. By designing the tutorial like this, I divide the difficult problem of solving the entire Rubik's cube into seven small problems which are easier for users to solve.

Another challenge for my app is to develop a solving algorithm that makes the computer capable of solving a Rubik's cube (For solution demonstration use). My approach to overcome the challenge is completely the same as provided in tutorial - teaching the computer how to solve the cube step by step.

Finally, I incorporate a cross solver in my app that uses a completely different algorithm from the one I use to solve the cube in the tutorial. The algorithm I use in the cross solver is the two-way-breadth-first search algorithm, which is a searching algorithm that searches for the optimal solution (as few steps as possible) to solve the cross. As you can see in the video, this algorithm I use is much more efficient than the one I use in the tutorial. The detailed implementation of the algorithm is in the file `Cross_Solver.py`.

About why I design my UI:

Prior to starting to do my project, I made a competitive analysis on my project. I found most tutorial programs separates the teaching mode and the solution demonstration mode. What I want to do is an integrated experience. (Tutorial based, however, you can see the solution at any stage during the tutorial.) I believe that by designing my UI like this I would give the users a much more educational experience.