Etude 4 Cubes Report

By Matthew Harvey and Carl Aiau

Introduction

We solved this etude by using brute force, calculating every possible combination of a 2x2x2 cube with two different colours. By using brute force, it guarantees that all combinations are considered so nothing is missed.

Mapping

To simulate the 2x2x2 cube, we created a 24 sized array and set 0 to blue and 1 to yellow. Each one of the 24 faces are mapped to the array, we used a real-life example just to visualise everything, which would be very useful with the rotation aspect.

Example of the mapping:

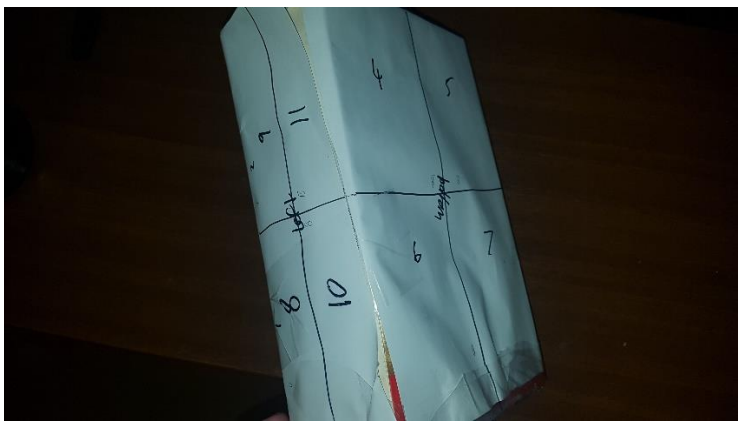
Top bottom left right front back

[0][0][0][0] [0][0][0][0] [0][0][0][0] [0][0][0][0] [0][0][0][0] [0][0][0][0]

which resembles a 2x2x2 cube fully blue.

The total number of combinations of blue and yellow faces on a 2x2x2 cube without any restrictions is 16777216.

The numbers in the picture represents the index of the array.

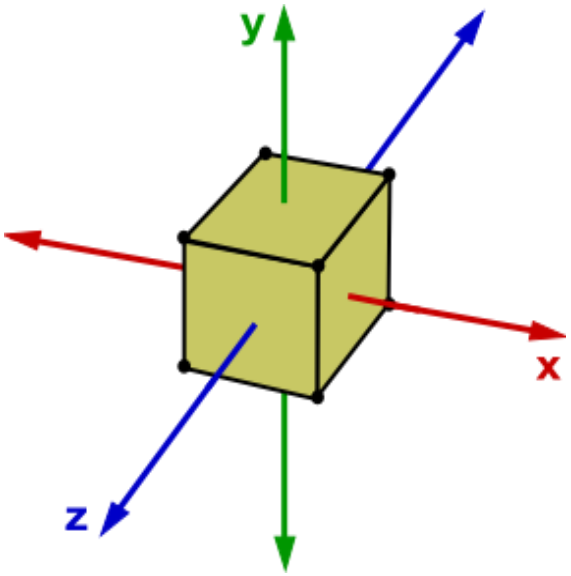


Valid Coloured Cube

Each 1x1x1 cube used in the 2x2x2 cube must have the same colour. Therefore, all the corners of the cube must be the same colour. So, for all the possible combinations, only arrays where specific colours are equal can be allowed, i.e. in the example face 11 and face 4 must have the same colour.

The total combination of 2x2x2 cubes with valid colour positions is 256.

Rotation



For rotation, the cube can spin on the x,y and z axis. Because it is square it must be rotated in 90 degree increments to check for any duplicated results in the 256 combinations remaining.

Our rotation worked like this,

Rotate z axis 3 times and check for duplicates.

Then, for the 3 90 degree x axis rotation's, rotate z axis 90 degrees 3 times.

Then, for the 3 90 degree y rotation's, rotate z axis 90 degrees 3 times.

The axis must be rotated 3 times to check the full 360 degree of the cube.

Rotations required to check for all axis combinations: 23

Rotations where programmed by using the mapped array and swapping values around based on their current indexed position. So from the picture a rotation may cause the value at 9,11,8 and 10 to move to position 4,5,6 and 7 which would resemble part of the cubes rotation.

For every rotation made, it is checked by a list of valid cubes to make sure there is no duplication for the counter. If no duplication, the cubes array is added to the list.

Summary

Because all valid rotations are compared to the solutions list, and every possible combination is considered, all possible unique combinations are found and counted, giving us the final answer.