Ideas

# Analyse:

Analyse position of each sticker and then work out location of each cube

Then use sticker location combined w/ cube location to work out rotation of cube and therefore steps needed to solve/perform step

# Steps

1. Analyse
   * When analyse clicked check all buttons changed
   * Then check if stickers are correct i.e. yellow/white etc. aren’t touching
   * Calculate cube pos’s from sticker locations
   * Calculate steps completed
     + If cross is complete (2, cross\_face.colour) else check new face, if all checked and false (1, white)
     + If face is complete check if sides are **not** complete (1, face.colour) else if complete check middle, if middle = false (3, completed\_face.colour)
     + If face and middle = complete check lower cross, if LC = false (4, completed\_face.colour)
     + Etc…
2. Cross
3. Corners
4. Middle
5. Lower Cross
6. Lower sides
7. Lower corners
8. Final

# Possible cubes

## Corners

1. W/R/B
2. W/B/O
3. W/O/G
4. W/G/R
5. Y/B/R
6. Y/R/G
7. Y/G/O
8. Y/O/B

## Sides

1. W/B
2. W/O
3. W/G
4. W/R
5. R/B
6. B/O
7. O/G
8. G/R
9. Y/B
10. Y/R
11. Y/G
12. Y/O

## Middle

1. W
2. R
3. B
4. O
5. G
6. Y

(3^3-1)

# Ideas

* When rotating assign the 9 values to a secondary array, then re-assign.
* Check number of each letter in Stickers() = 9
* To check order of stickers on corners:?
  + Find white/yellow corner?
  + Work clockwise?
  + …?
* Class for cube?
  + Type?
  + Location in 3d array (3\*3\*3)
  + Rotation?
  + Colours?
    - Therefore 26 (27 if void/middle counted) cube objects in total