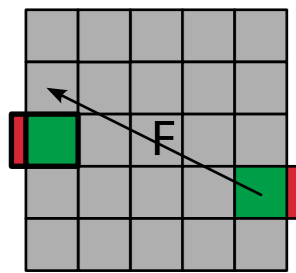
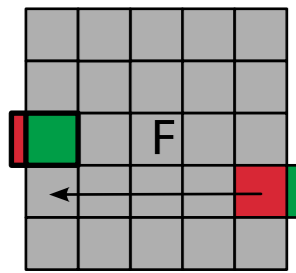


- \* now 4 correct edges are in white layer, and 4 correct edges are in yellow layer
- \* for the last 4 (vertical) edges, first repair the middle layer centers so all centers are correct by using **u, E & d**
- \* from now on, side faces turn only 180° (not 90°) or use *edge flipping algorithm*
- \* look at the edge in the front-left side, look at the middle edge piece, this is leading (in this case green/red)
- \* put a matching edge piece on the front-right edge only use: **R2 or B2**



[1] if front colors matches

**u' flip u**



[2] if front colors don't match

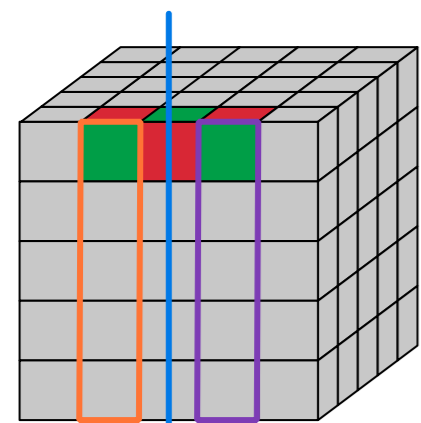
\* first flip the edge

**flip**

\* then put in place

**d flip d'**

- \* if only one edge is incorrect, because some of the pieces have to be flipped (in this case, two of the three edges have to be flipped) use this algorithm to solve *the parity*



put the edge in front-top

**r'U<sup>2</sup> lF<sup>2</sup> l'F<sup>2</sup>  
r<sup>2</sup>U<sup>2</sup> rU<sup>2</sup> r'U<sup>2</sup>  
F<sup>2</sup>r<sup>2</sup>F<sup>2</sup>**

slice 2 slice 1

symmetric

in this case: **r**

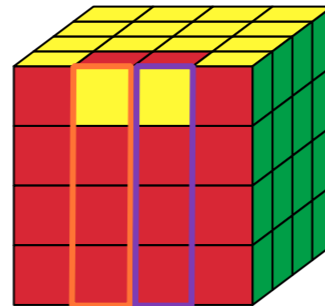
in this case: **l**

(you can also turn both **l+l** instead of only **l** or turn both **r+r** instead of only **r**)

### 3) solve like 3x3

- \* now all centers are correct and all edges are paired, the cube can be solved like a normal 3x3 cube
- \* however there can be some parity (some edge pieces seem to be correct, but actually they need to be swapped with other edge piece with same colors)
- \* parity if n (number of layers) is even

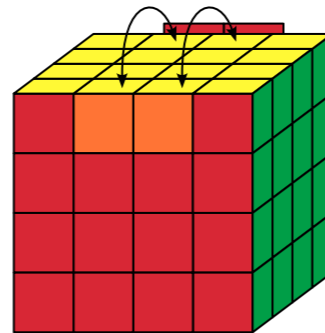
[1] one edge need to be flipped



slice 2 slice 1

\* use algorithm described in previous column

[2] 2 of the 4 top edges have incorrect position

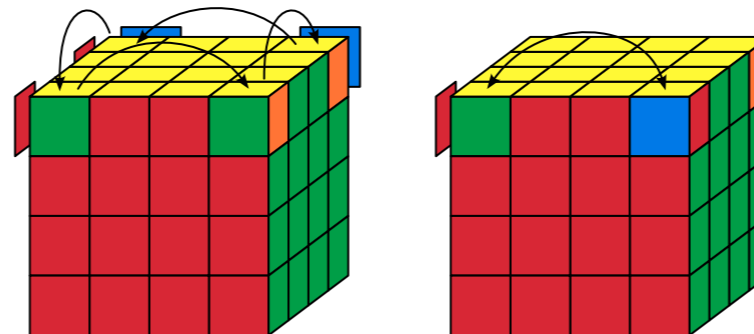


\* swap places of top-front and top-back edge

**r<sup>2</sup>U<sup>2</sup> r<sup>2</sup>(U+u)<sup>2</sup> r<sup>2</sup>u<sup>2</sup>**

\* use 3x3 algorithms to solve

[3] corners are well orientated, but can't be placed correctly using 3x3 algorithms



\* use algorithm [2] to swap two edges

\* use 3x3 algorithms to solve