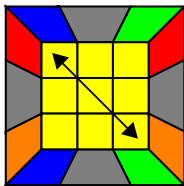
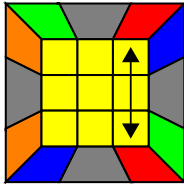


Rubik cube solving, COFP method: 2-Look PLL

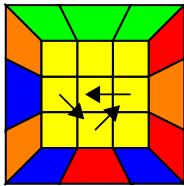
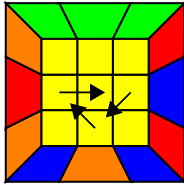
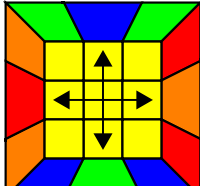
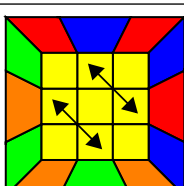
0 Moves

M: slice move, follows the L direction

1 Corners

Name	Case	Algorithm
Diagonal: all different ↻ ↻, ↻ ↻		$F (R U' R') U' (R U R') F'$ $(R U R') U' (R' F R F')$
Headlights ↻ ↻, ↻ ↻		$(R U R') U' (R' F R F')$ $F (R U' R') U' (R U R') F'$

2 Edges

Name	Case	Algorithm
Right center to left		$R U' (R U R U R) (U' R' U' R^2)$
Left center to right		$L' U (L' U' L' U' L') (U L U L^2)$
Opposite edge swap		$M^2 U M^2 U^2 M^2 U M^2$
Adjacent edge swap		$M' U M^2 U M^2 U M' U^2 M^2$

Source: <https://jperm.net/algs/2lookpll>

Rubik cube solving, COFP method: 2-Look PLL

3 Corners training scrambles

- Diagonal:
 - initial: B F U2 B F' U' R2 U L2 D' L2 U' F2 D L2 D' R2
 - repeat: U' F B' U2 F' B U2 L2 F2 U F2 U2 L2 U' F2 U2 F2 L2
- Headlights:
 - initial: U' R L U2 R L' U F2 L2 B2 L2 F2 D' R2 D B2 U' R2
 - repeat: B F U2 B F R2 U F2 L2 D B2 U' F2 R2 D' R2 B2 F2

4 Edges training scrambles

- Right center to left: U2 R2 U' F2 R2 F2 U R' L F2 D2 B2 U2 R L B2 D2
- Left center to right: U2 B2 U R2 B2 R2 U R2 U2 R2 U2 B2 R2 B2 U2 B2
- Opposite edge swap: U F B D2 F B D2 F2 R2 D2 R2 B2 L2 U2 F2 L2
- Adjacent edge swap: U R L U2 R L' U B2 R2 B2 U R2 U B2 R2 B2 U R2

5 Adjacent Edge Swap Parity on 4x4

- R U R' U (parity) U R U' R'
- Parity: r2 U2 r2 Uw2 r2 Uw2