

F2L ALGORITHMS

FOR PERSONAL USE ONLY

CONTENTS:

Page 1- Information

Page 2- Corner Facing Front in First Layer

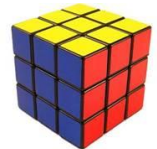
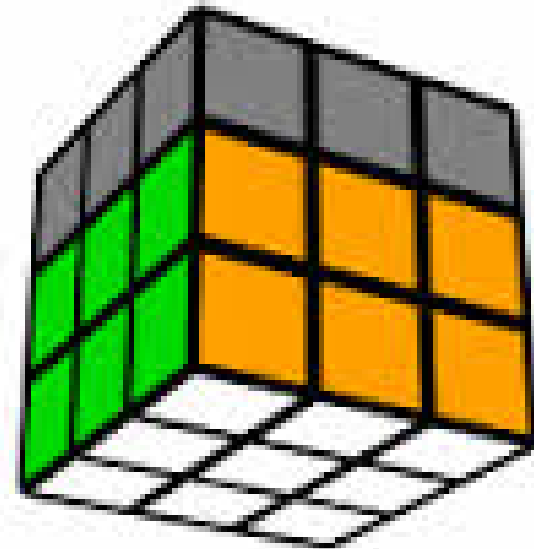
Page 3- Corner Facing Front in Last Layer

Page 4- Corner Facing Right in First Layer

Page 5- Corner Facing Right in Last Layer

Page 6- Corner Facing Up in Last Layer

Page 7- Corner Oriented in First Layer



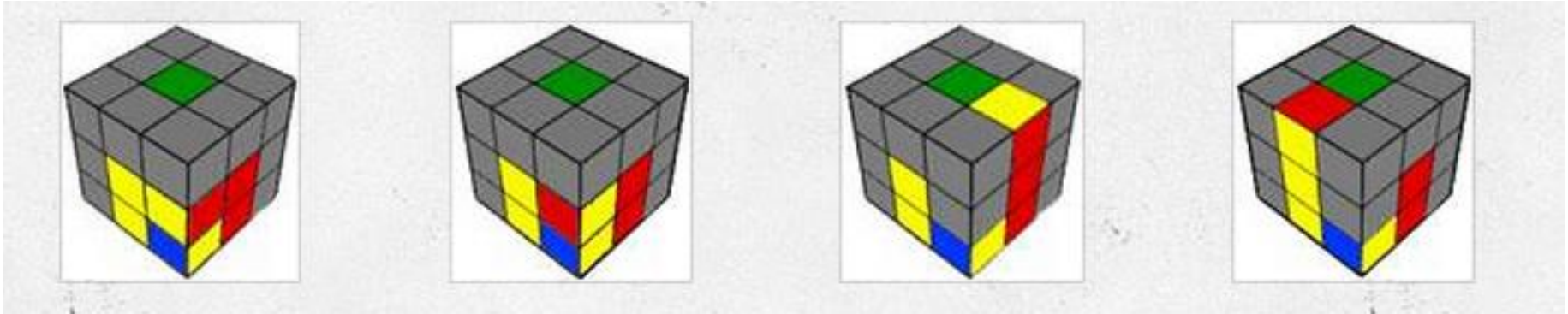
FSCUBING

Information

F2L is the second step in the CFOP Method. F2L stands for First 2 Layers, which is better than the less efficient beginners method, where you do them both individually. This is a really efficient way to complete the First 2 Layers, as it will give you a better time, knocking off about 5-10 seconds off of your previous times, where you used the beginners method. F2L can be learnt intuitively, which is easier to learn, but can take more time. At this page, you will learn set algorithms for most F2L cases. These pages will not give you all of them, as there are certain cases which you may be able to figure out with intuition. Once you have memorised all of these cases, I recommend that you practice each of them individually, so that you have a fast, finger trick-able method of solving each of these cases. There are 42 Cases displayed here on these pages. I have presented the algorithms with the standard algorithms underneath them. There are many more ways of solving each case, which probably won't be displayed on this page. You will have to find the quickest algorithm yourself, if you don't prefer the specified algorithm. If you can think of any more algorithms, please use the contact page on the website to give me the algorithm, so we can display it on this page. Thank You!



Algorithms- Corner Facing Front in First Layer



Case #1

Case #2

Case #3

Case #4

Case #1- R, U, R', U', R, U², R', U', R, U, R'

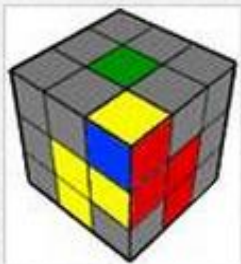
Case #2- R, U', R', d, R', U', R, U', R', U', R

Case #3- R, U', R', U, R, U', R'

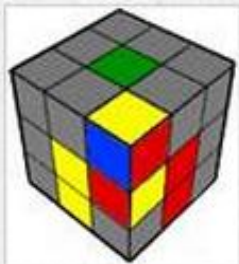
Case #4- (y'), R', U', R, U, R', U', R



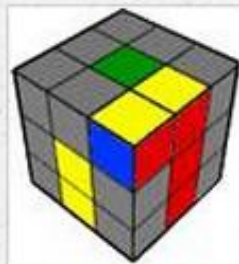
Algorithms- Corner Facing Front in Last Layer



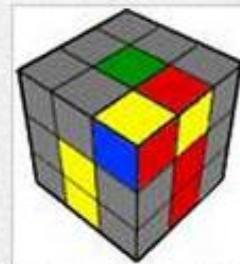
Case #1



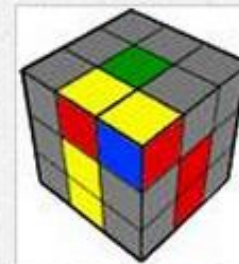
Case #2



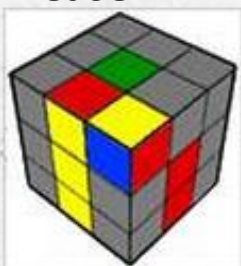
Case #3



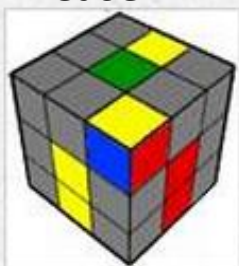
Case #4



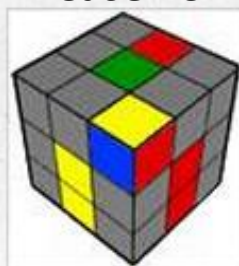
Case #5



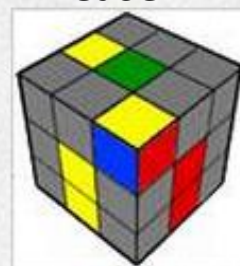
Case #6



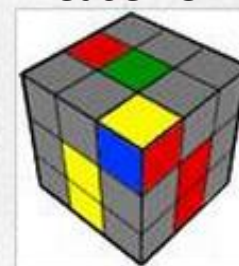
Case #7



Case #8



Case #9



Case #10

Case #1- U', R, U', R', U2, R, U', R'

Case #2- U', R, U, R', d, R', U', R

Case #3- U, R, U', R'

Case #4- d', L', U, L, U', L', U', L, U, L', U', L

Case #5- F', U, F, U2, R, U, R'

Case #6- d, R', U, R, U', R', U', R

Case #7- U', R, U, R', U2, R, U', R'

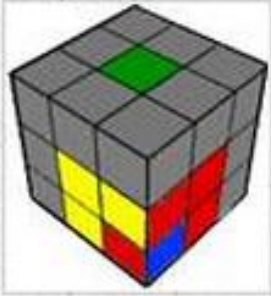
Case #8- d, R', U', R, U', R', U', R

Case #9- U, R, U2', R', U2, R, U', R'

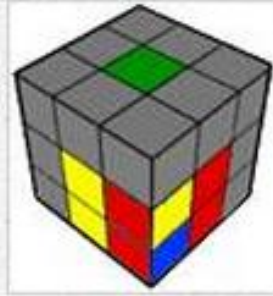
Case #10- (y') R', U', R



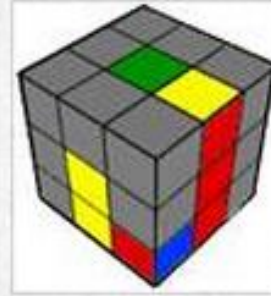
Algorithms- Corner Facing Right in First Layer



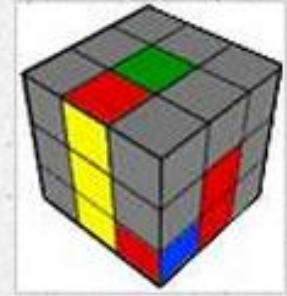
Case #1



Case #2



Case #3



Case #4

Case #1- R, U2', R, U, R', U, R, U2', R2'

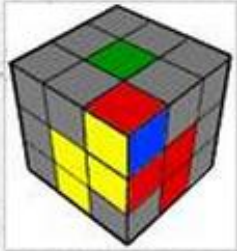
Case #2- R, U', R', U2, (y'), R', U', R, U', R', U, R

Case #3- R, U, R', U, R, U, R'

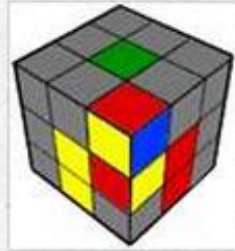
Case #4- (y') R', U, R, U', R', U, R



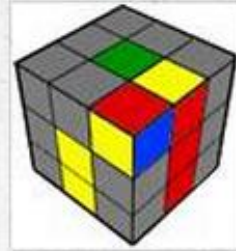
Algorithms- Corner Facing Front in Last Layer



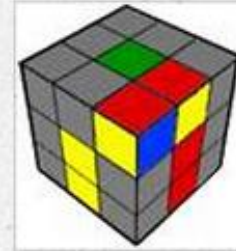
Case #1



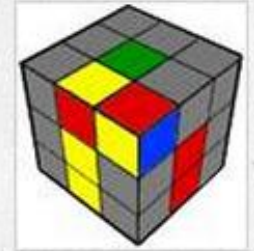
Case #2



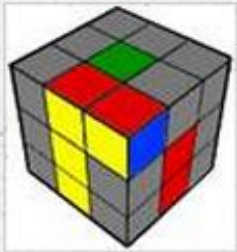
Case #3



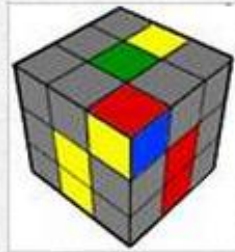
Case #4



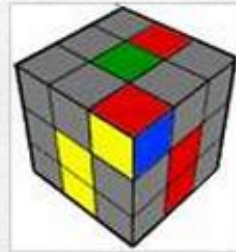
Case #5



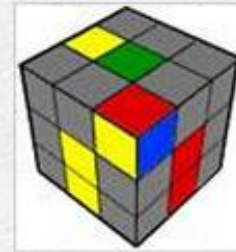
Case #6



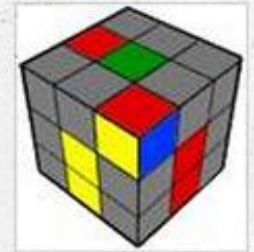
Case #7



Case #8



Case #9



Case #10

Case #1- d, R', U, R, U2', R', U, R

Case #2- d, R', U', R, d, R, U, R'

Case #3- U', R, U', R', U, R, U, R'

Case #4- R, U', R', U2, (y'), R', U', R

Case #5- U, R, U', R', U', R, U, R', U', R, U, R'

Case #6- d', L', U, L

Case #7- R, U, R'

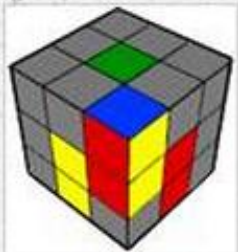
Case #8- d, R', U2, R, U2', R', U, R

Case #9- U', R, U, R', U, R, U, R'

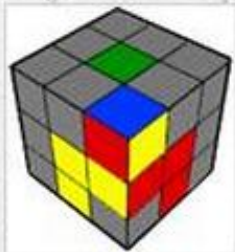
Case #10- d, R', U2, R, U2', R', U, R



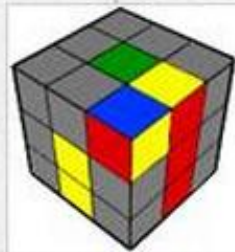
Algorithms- Corner Facing Up in Last Layer



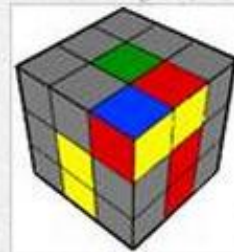
Case #1



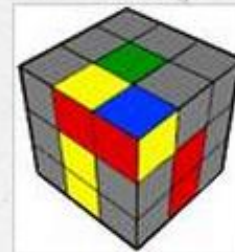
Case #2



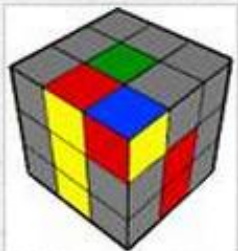
Case #3



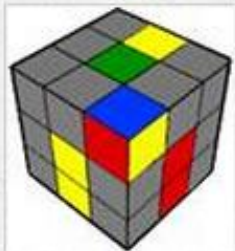
Case #4



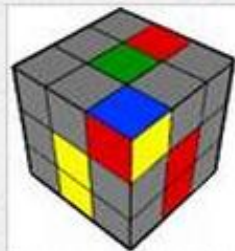
Case #5



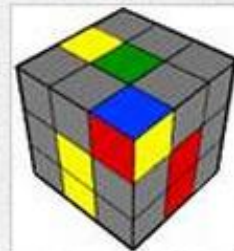
Case #6



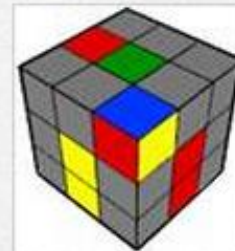
Case #7



Case #8



Case #9



Case #10

Case #1- R, U', l, U', R', U, l'

Case #2- R, U, R', U', R, U, R', U', R, U, R'

Case #3- R, U2, R', U', R, U, R'

Case #4- (y'), U2, R2', U2, R, U, R', U, R2

Case #5- U2, R2', U2, R', U', R, U', R2'

Case #6- (y'), R', U2, R, U, R', U', R

Case #7- U, R, U2, R', U, R, U', R'

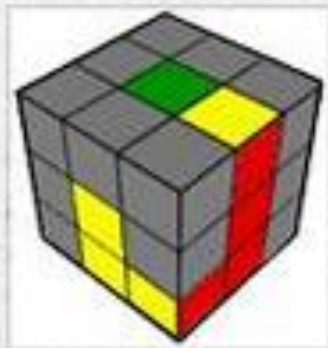
Case #8- d2, (y), R', U', R, U', R', U, R

Case #9- U2, R, U, R', U, R, U', R'

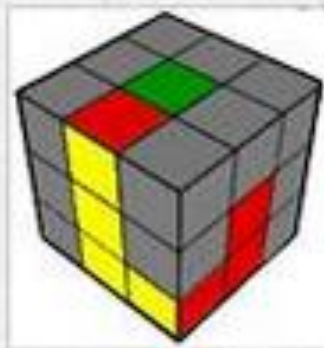
Case #10- d', L', U2', L, U', L', U, L



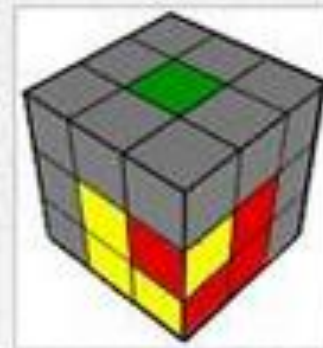
Algorithms- Corner Oriented in First Layer



Case #1



Case #2



Case #3

Case #1- R, U', R', U', R, U', R', U, R, U, R'

Case #2- U, R, U', R', d', L', U, L

Case #3- R, U', R', d, R', U2, R, U2', R', U, R

