4x4x4-Cube lösen (Beginners method)

Hintergrund-Infos und Vorbemerkungen:

Diese Anleitung richtet sich an durchschnittlich talentierte Menschen wie mich ;-).

Sie setzt dabei aber zwingend voraus, dass man den regulären Zauberwürfel (also den 3x3-Rubiks Cube, im Speedcube-Slang abgekürzt zu: ”3x3”) problemlos nach der Beginners- oder sonst einer Methode lösen kann.
(Manchmal liest man auch von 3x3x3 — bzw. analog: 4x4x4: Die Zahl bezieht sich auf die Anzahl der sichtbaren Cubies für jede der 3 Dimensionen X, Y, Z.)

Ebenfalls von Vorteil ist es, wenn man komplexere Cubes, wie den Fisher-Cube lösen kann - das Konzept von Parity-Cases wird einem dann sofort einleuchten bzw. bekannt vorkommen!

Noch ein interessanter Aspekt: Mathematisch lassen sich Zauberwürfel der Gruppentheorie zuordnen. Während jedoch ein normaler 3x3x3 ”nur” etwa 43 Trillionen verschiedene Zustände haben kann, ist dies beim 4x4x4 eine deutlich höhere Zahl, nämlich:

7.401.196.841.564.901.869.874.093.974.498.574.336.000.000.000

oder circa

7.401.196.841.564.901.869.874.093.974 Trillionen

oder eben circa

7,4 Septilliarden

eine alptraumhaft große Zahl um per zufälligem Drehen (”Brute Force”) zur Lösung zu kommen, oder?!?

Ein wichtiger Tipp: Vor dem Verdrehen: *Farbschema ermitteln!*

Speedcuber wissen in der Regel, dass es den regulären ”3x3x3” in verschiedenen Farbschemata gibt: So haben asiatische Würfel oft andere Farben als die europäischen. Bevor man also den 4x4x4 verdreht, empfiehlt es sich sehr, zu prüfen, ob das Standard-Farbschema gilt. Das heißt: Wenn die Oberseite des 4x4x4 gelb ist, dann sollte auf der Würfelunterseite weiß sein. Und wenn frontal die orangene Farbe sichtbar ist, dann erwarten wir, dass rot auf der Rückseite ist. Und ebenfalls ausgehend von orangener Frontalfarbe, sollte auf der rechten Würfelseite die blaue Farbe sichtbar sein. Und gegenüber davon, auf der linken Würfelseite sollte sich die grüne Seite befinden. Ist das der Fall, dann haben wir das europäische Standard-Farbschema. Das müssen wir auswendig lernen, damit wir den einzelnen Seiten keine falschen Farbkombinationen zuweisen und der Würfel damit unlösbar bleibt. (Dieser Fall ist beim 3x3x3 konstruktionsbedingt ausgeschlossen, da sich die Farben der Würfelseiten unabänderlich aus den Mittelsteinfarben ergeben - und die hängen zueinander fix an einem drehbaren Kreuzgelenk im Innern des Cubes. Man sagt auch: Die Mittelsteinfarben sind relativ zueinander starr.) Das ist beim 4x4x4 anders: Hier kann man beispielsweise zuerst die weißen Mittelsteine lösen und danach auf der angrenzenden Würfelseite die gelben Mittelsteine. Nur wird man dann merken, dass sich die beiden Würfelseiten nicht lösen lassen, weil die vorhandenen acht Ecksteine, diese illegale Farbkombination nicht hergeben / unterstützen / abbilden. Ergo: Man muss das Farbschema im Kopf haben.

Hat man den Würfel schon verdreht, so kann man die Farbverteilung des gelösten Cubes trotzdem noch bestimmen. Dazu schaut man sich die acht Ecksteine (Corners) an und notiert, welche Farbkombinationen die Ecksteine haben. Durch Ausschlussverfahren kann man dann die gegenüberliegenden Farben herausbekommen.

Anmerkungen zur Nomenklatur von Algorithmen:

Großbuchstaben bedeuten, dass man nur die entsprechende Würfelseite dreht, sobei Deckeldrehung mit dem Buchstaben ”O” (wie ”Oben”), die Drehung der Rückseite mit ”B” (”Back”), Frontaldrehungen mit dem Buchstaben ”F”, die Drehung des Würfelbodens mit ”U” (wie: ”Unten”) und die Drehung der rechten und linken Würfelseite mit ”R” und ”L” gekennzeichnet sind. Kleine Buchstaben besagen, dass man die beiden äußeren Layers gemeinsam drehen soll, z. b. ”r” soll das 3. und 4. vertikale Layer (von links her gezählt - faktisch also die beiden rechten Layers) bezeichnen. Diese sind dann gemeinsam zu drehen. Man nennt diese Drehungen auch: *”broad moves”*.

Links hochgestellte arabische Ziffern: Sie zeigen an, wieviel 90-Grad-Drehungen gemeint sind. Beispielsweise bedeutet ${}^{2}r\downarrow$, dass man die beiden rechten vertikalen Layers um 180 Grad nach vorne herunterklappt.

Rechts tiefgestellte römische Zahlen zeigen, welches vertikale Layer von links her gemeint sind. Entsprechend bedeutet $R_{III}\uparrow$, dass das dritte von links her zu zählende, vertikale Layer um 90 Grad (also eine Vierteldrehung) nach oben zu schieben ist.

Wie gehen wir vor? - Das allgemeine Kochrezept

Ganz allgemein gesprochen folgen wir dem Ansatz der Hoya-Methode, die im Gegensatz zur Methode nach Robert Yau zuerst alle vier Centers zusammenbaut und dann mit dem weißen Kreuz und dann OLL und PLL fortfährt, und damit — bildlich gesprochen — den 4x4x4 Cube auf einen 3x3x3 Cube reduziert.

Die Idee ist also, dass wir alle 4 Mittelsteine jeder Seite gleichfarbig machen und die eizelnen Kantensteine zu gleichfarbigen Paaren ordnen. Wenn das der Fall ist, können wir die 4 Mittelsteine auf jeder Seite als einen Mittelstein in der entsprechenden Farbe ansehen. Und analog dazu können wir auch jedes der Kantenpaare als einen einzigen Kantenstein umdeuten. Wenn wir das tun, können wir den 4x4x4-Würfel grundsätzlich nach der normalen 3x3x3-Beginners-Methode lösen.

Im Unterschied zum normalen 3x3x3 können dabei aber sowohl bei OLL als auch bei PLL Parity-Cases auftreten, die wir gesondert behandeln und auflösen müssen.

Wir gehen also beim Lösen stufenweise vor:

- Lösen der 4 Mittelsteine auf einer (z. B. der weißen) Seite = Würfelunterseite
- Lösen der 4 Mittelsteine auf der gegenüberliegenden Seite (z. B. gelb) = Würfeloberseite
- Lösen der 4 Mittelsteine auf einer als Frontalseite zu wählenden Farbe (z. B. orange)
- Lösen der 4 Mittelsteine auf der zur eben gewählten Frontalseite korrespondierenden Rückseite (z. B. rot)
- Lösen der 4 Mittelsteine auf der zur eben gewählten Frontalseite korrespondierenden rechten Würfelseite (z. B. blau)
- Lösen der 4 Mittelsteine auf der zur eben gewählten Frontalseite korrespondierenden linken Würfelseite (z. B. grün)
- Verheiraten der Kantenpaarsteine = alle Kantenpaare farblich passend nebeneinander stellen
- Auf der zur eben gewählten Frontalseite am Boden befindliche Würfelunterseite: Weißes Kreuz aufbauen
- Weiße Ecksteine ausrichten bzw. lösen → Weiße Grundseite auf der Würfelunterseite fertig stellen
- Einsetzen der Kantenpaare an ihre Home-Position → erreiche Status F3L.
- Gelbe Linie, ggf. Parity-Behandlung, gelbes Kreuz → erreiche Status OLL.
- Falls aufgetreten: Weitere Parity-Cases beseitigen
- Wüfel komplett fertigmachen (PLL)

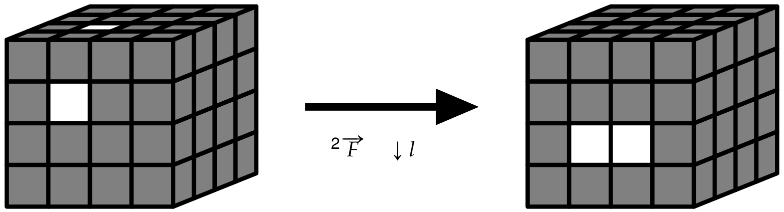
Lösen der 4 weißen Mittelsteine:

Während es beim 3x3x3 Cube nur einen Mittelstein gibt, haben wir auf jeder Würfelseite 4 Mittelsteine, die wir auf jeder Würfelseite gleichfarbig machen müssen. Prinzipiell kann man mit jeder Farbe zu lösen beginnen — solange man das Farbschema des Würfels einhält. Jedoch wollen wir hier grundsätzlich mit weiß beginnen, um einen klaren Fahrplan zu haben, auch wenn man später mit mehr Übung entdeckt, dass das Beginnen mit einer anderen Farbe vielleicht Vorteile bringen kann.

Grundsätzlich versuchen wir im Regelfall folgendes zu tun:

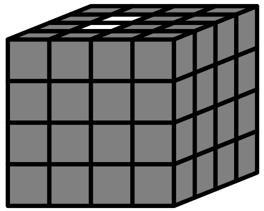
1. Wir suchen auf dem Cube Mittelsteine, die die Farbe weiß haben.

2. Finden wir zwei solche Mittelsteine stellen wir sie direkt nebeneinander ("wir paaren die Mittelsteine") - es entsteht die erste weiße Zweierbar. Das Paaren geschieht dabei völlig intuitiv - es gibt kein Standardalgorhithmus dafür! - es ist andererseits aber auch nicht sehr schwer, das erste Paar zu bilden. Steht oben am Deckel beispielsweise eine weiße Fläche und frontal auch, so drehen wir die Front so, dass die weiße Fläche beim Herunterdrehen neben der weißen Fläche an der Front zum Liegen kommt:



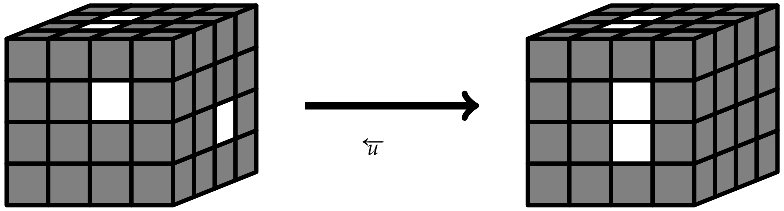
(Hier sind die nichtrelevanten Steine grau eingefärbt. Man sieht auf dem Bild links, dass man zuerst die Frontseite zweimal im Uhrzeigersinn drehen muss, um danach die 2. Vertikalebene von links nach unten drehen zu können. Damit entsteht dann das erste Zweierpaar. Tipp: Um Slicedrehungen zu vermeiden, kann man auch die beiden linken Vertikalebenen zusammen herunterdrehen.)

3. Gedanklich definieren wir: Dort wo die Zweierbar ist, ist der Würfeldeckel! Hierzu drehen wir den Würfel global so, dass unser Zweierpaar oben ist und sozusagen den neuen Deckel bildet. Später werden wir diese Arbeitsdefinition wieder fallen lassen und gelb als Deckel setzen - damit ist dann alles wie beim normalen 3x3x3 weiterzulösen...aber vorerst ist weiß noch oben:



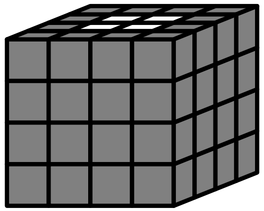
4. Wir versuchen auf einer (relativ zum Würfeldeckel zu sehenden Würfelseite eine zweite weiße Zweierbar aufzubauen. Dabei müssen wir achtgeben, dass die schon gelöste weiße Zweierbar am Würfeldeckel nicht durch unbedachte Drehungen wieder kapputt gemacht wird. **Aus diesem Grunde sind (zumindest vertikale) Slice-Drehungen ab sofort als kritisch zu betrachten, da sie eine innewohnende Tendenz haben, gelöste Mittelsteine in einen unsolved state zu bringen!**

5. Auch hier gehen wir intuitiv vor und bauen die zweite Zweierbar mit Hilfe von Frontdrehungen und Kombinationen von u-Drehungen (die ersten beiden Layers des Würfelbodens) auf. Gleichzeitig achten wir bei den Drehungen darauf, dass die Zweierbar am Würfeldeckel erhalten bleibt. Sobald die zweite weiße Zweierbar auf der Würfelseite gebildet worden ist, richten wir sie so aus, dass sie wie eine vertikale Linie auf der Würfelseite sichtbar ist:



Da wir noch auf nichts sonst Rücksicht nehmen müssen, können wir einfach die beiden Zweierbars vereinigen, indem wir die beiden rechten Ebenen mit einem r-Move vertikal nach oben ziehen, also: r↑

Im Ergebnis haben wir unsere ersten 4 weißen Mittelsteine beieinander, sozusagen im ***solved state***:



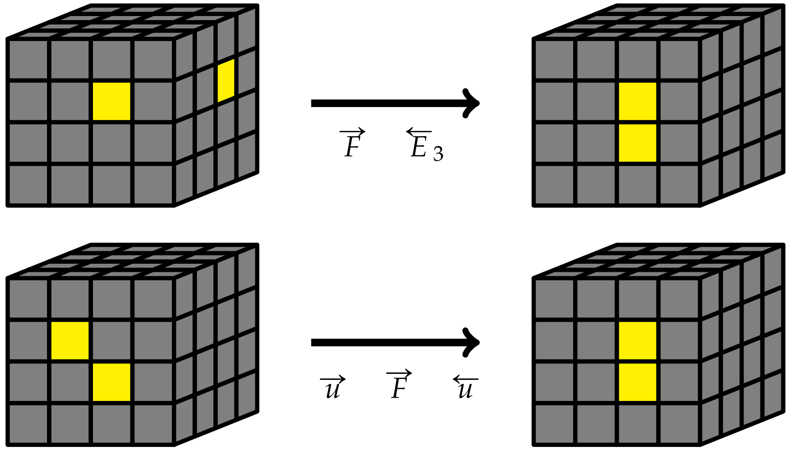
Lösen der 4 gelben Mittelsteine

Da die gelbe Würfelseite der weißfarbigen Würfelseite gegenüberliegt, geben wir unsere bisherige Arbeitsdefinition auf und halten ab jetzt den Würfel so, dass am Würfelboden die weißen Mittelsteine sind (diese definieren ja die Farbe der Seite). Unser Ziel ist es dann, die vier gelben Mittelsteine auf dem Würfeldeckel nebeneinander zu platzieren.

Das Vorgehen ist vom Prinzip her dasselbe wie bei den weißen Mittelsteinen: Wir bilden nacheinander zunächst die erste gelbe Zweierbar, dann die zweite und dann verheiraten wir beide Zweierbars auf derjenigen Seite, die den weißen Mittelsteinen gegenüber liegt. Der Unterschied besteht allerdings in der Ausführung, denn wir haben nicht mehr so viele Freiheitsgrade wie beim Lösen der vier weißen Mittelsteine: Auf diese müssen wir immer Acht geben, damit sie nicht in einen *"unsolved state"* geraten.

Wie gehen wir nun vor?

Zunächst bauen wir — auch wieder intuitiv — auf einer Würfelseite eine gelbe Zweierbar. Liegen z. B. gelbe Flächen auf der gleichen Ebene bzw. auf dem gleichen Layer, dann können wir durch eine rechte Frontdrehung, gefolgt von einer Equator-Layer3-Slicedrehung die gelben Flächen zusammenschieben. Beispielsweise:

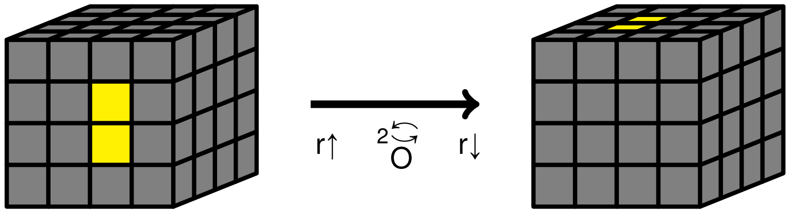


Nun können wir natürlich nicht einfach die gelbe Zweierbar "hochziehen", sonst würden die weißen Mittelsteine am Boden auseinandergerissen. Daher brauchen wir einen Algorithmus, der die Zweierbar gefahrlos hochbringt.

Der Hochbring-Algorithmus:

r↑ ^{2}O r↓

Der Hochbring-Algorithmus besteht aus drei Teilen: Der erste Move **(Setup-Move)** bringt die gelbe Zweierbar zum Deckel (wodurch temporär die weiße Mittelsteinordnung am Würfelboden zerstört wird). Der zweite Move verschiebt die gelbe Zweierbar auf dem Deckel und bringt sie sozusagen in Sicherheit **(Sicherungsmove)**. Der dritte Move **(Setback-Move)** macht das Gegenteil des ersten Moves (er nimmt ihn sozusagen zurück) und heilt damit am Boden die zerstört gewesene Mittelsteinordnung.



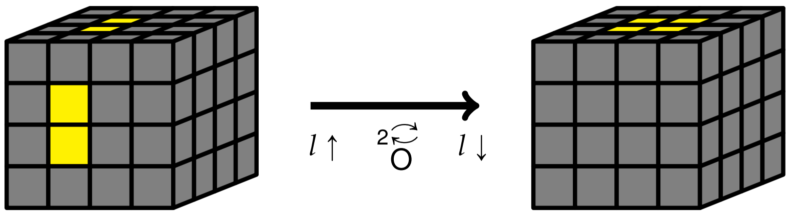
Im Ergebnis haben wir damit unsere erste gelbe Zweierbar am Deckel stehen. Nun versuchen wir wieder, auf den Würfel-Seitenflächen (also nicht oben und nicht unten) die zweite gelbe Zweierbar mit Hilfe von gefahrlosen F-Drehungen, R-Drehungen, L-Drehungen, B-Drehungen, O-Drehungen, U-Drehungen und E-Drehungen intuitiv hinzubekommen. Vermeiden sollte man dagegen alle broad-Drehungen (mit Ausnahme von o-Drehungen), da sie potentiell die weiße Mittelsteinordnung zerstören können. Auch vertikale Slices sind gefährlich (hier jedoch noch nicht: horizontale Slices bzw. Equator-Drehungen, die wir mit dem Buchstaben E bezeichnen.)

Haben wir unsere zweite gelbe Zweierbar seitlich aufgebaut, so richten wir sie durch F-Drehungen so aus, dass sie erstens vertikal ist und zweitens mit der gelben Zweierbar am Deckel eine gedankliche Linie (unterbrochen durch die Kantensteine) bildet, also beispielsweise:

Wenn wir dies geschafft haben, können wir die beiden Zweierbars paaren, indem wir den Hochbring-Algorithmus von gerade eben anwenden. Aus dem Hochbring-Algorithmus, wird dann sozusagen der *kurze Paarungsalgorithmus* — wir müssen dabei nur aufpassen, dass ihn richtig herum ausführen. Warum "kurz"? — Damit ist gemeint, dass die Linien direkt auf benachbarten Seiten stehen. Weiter unten werden wir noch einen *langen Paarungsalgorithmus* einführen, wenn sich die gleichfarbigen Zweierbars auf gegenüber liegenden Seiten (opposite) befinden. Dazu später mehr.

Linksseitiger kurzer Paarungs-Algorithmus:

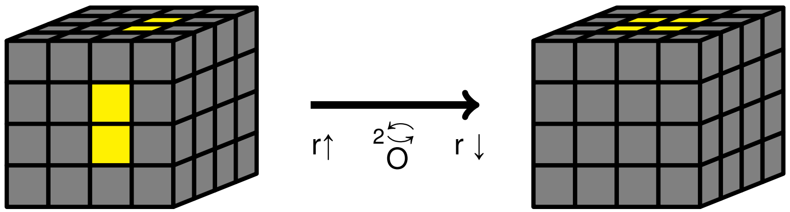
↑ *l* ^{2}O ↓ *l*



Wie aus der vorherigen Seite aus dem Bild oben rechts hervorgeht, kann auch eine zweite (gespiegelte) Siutation vorliegen. Dann müssen wir den Paarungsalgorithmus auf diese Situation hinbiegen:

Rechtsseitiger kurzer Paarungs-Algorithmus:

$$r\uparrow \quad \overset{2\circlearrowleft}{\mathrm{O}} \quad r\downarrow$$



Haben wir die gelben Mittelsteine am Würfeldeckel zusammengeführt, ohne die Ordnung der weißen Mittelsteine am Würfelboden zu zerstören, **kippen wir den Würfel erst einmal global nach rechts,** so dass die vier gelben (solved-) Mittelsteine nun rechts und die vier weißen (solved-) Mittelsteine nun links sind. Der Sinn dahinter ist, dass wir auf diese Weise viel weniger versucht sind, die weiße und gelben Mittelsteinordnung erneut zu zerstören.

Lösen der 4 orangefarbenen Mittelsteine:
Auf die gleiche Art und Weise wie gerade gezeigt, verfahren wir im nächsten Schritt:

Wir bilden mir r-Moves, l-Moves und F-Moves (wie schon dargestellt) nacheinander zwei orangefarbene Zweierbars und verheiraten sie am temporären Würfeldeckel. Ist das geschafft, drehen wir den Würfel auf den Kopf, um die gegenüberliegende Seite zu bearbeiten.

Lösen der 4 roten Mittelsteine:
Da rot gegenüber von orange ist, haben wir gerade den Würfel auf den Kopf gedreht, so dass rot oben liegt. Erneut wenden wir dasselbe Hochbring- und Paarungs-Prinzip an. Falls sich nach dem Aufbau der beiden Zweierbars ergibt, dass sie nicht auf angrenzenden Würfelflächen sind, sondern sich die rote Linie auf gegenüberliegenden Seiten ergibt müssen wir unseren Paarungsalgorithmus etwas anpassen und den "langen Paarungsalgorithmus" ausführen:

Linksseitiger langer Paarungs-Algorithmus:

$$^{2}\uparrow l \quad \overset{2\circlearrowright}{\mathrm{O}} \quad ^{2}\downarrow l$$

Rechtsseitiger langer Paarungs-Algorithmus:

$$^{2}r\uparrow \quad \overset{2\circlearrowleft}{\mathrm{O}} \quad ^{2}r\downarrow$$

Sind die roten Mittelsteine am Würfeldeckel gelöst, **kippen wir erneut den Würfel global nach rechts.** Dadurch wird blau am Würfeldeckel zu beareiten sein.

Lösen der 4 blauen Mittelsteine, danach der 4 grünen Mittelsteine:
Wie gehabt lösen wir die blauen Mittelsteine, drehen den Würfel auf den Kopf und lösen die grünen Mittelsteine.

Tipp: Wir können dabei auch vertikale 180-Grad-Slices (also: R_1- und R_2-Moves bzw. L_1- und L_2-Moves — je nach Betrachtungsweise — (es ist faktisch dieselbe Drehung in unterschiedlicher Notation...je nachdem, ob ich die Vertikalslices von links oder rechts betrachte...), und dann den Deckel drehen und danach die anfängliche Slicedrehung wieder um 180 Grad zurückdrehen und damit auflösen.
Das Vorgehen ist sozusagen: *Setup* → *Rotate* → *Setback.*

Wichtig ist halt, dass alle bisher gelösten Mittelsteine am Ende in ihrer Ordnung bleiben (kontrollieren!) Temporär dürfen sie natürlich durch Moves zerstört werden — wenn es temporär bleibt!

Farbschema kontrollieren, Unstimmigkeiten beseitigen!

Nun betrachten wir einmal unser bisheriges Werk. Wir **müssen!** einmal kurz kontrollieren und sicherstellen, dass das Farbschema stimmt, sonst wird sich der Würfel nicht lösen lassen. Für europäische Würfel mit Standardfarbschema gilt das im entsprechenden Kapitel schon Gesagte: Wenn weiß unten ist, ist oben gelb. Wenn dann vorne orange ist, ist an der Würfelrückseite rot. Und außerdem ist dann blau auf der rechten Seite und grün auf der linken.

Das testen wir einmal kurz:

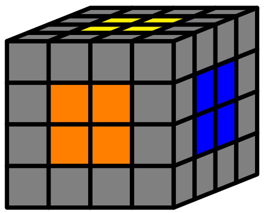
Denn es kann sein, dass wir beim bisherigen Lösen irgendwo einen Fehler begangen haben, so dass sich zwar die Opposite-Farben (rot:orange bzw. grün:blau bzw. gelb:weiß) gegenüberliegen, aber dass beispielsweise bei orangener Frontalseite (die Mittelsteinfarbe gibt ja die Würfelseitenfarbe vor), blau auf der linken Würfelseite ist und nicht rechts. Mit anderen Worten: Die adjazent (seitlich) liegenden Farben passen nicht zueinander! Dann müssen wir das korrigieren. Und auch nur dann führen wir den nachfolgend dargestellten Korrekturalgorithmus aus.

Korrekturalgorithmus für falsche adjazente Mittelsteinfarben:
Wir drehen den Würfel so, dass gelb oben, weiß unten und frontal blau steht. Wir wollen jetzt blau mit grün vertauschen, damit die Adjazentfarben passen. Dies geht wie folgt:

$$({}^2r\uparrow \quad {}^2\overset{\circlearrowleft}{O} \quad {}^2r\downarrow) \qquad ({}^2\uparrow l \quad {}^2\overset{\circlearrowright}{O} \quad {}^2\downarrow l)$$

Vorsicht Fehlerfalle: Die r- und l-Drehungen müssen um 180 Grad stattfinden, nicht um 90 Grad...

Zwischenergebnis: Alle Mittelsteine sind korrekt gelöst:



Verheiraten der Kantenpaare

Da es jeden Kantenstein exakt zweimal (völlig gleich aussehend) gibt, können wir diese als Kantenpaare bezeichnen. Unser Ziel ist es, die 12 vorhandenen Kantenpärchen so anzuordnen, dass sie direkt nebeneinander stehen. Korrekt nebeneinander stehende Kantenpaare bezeichnen wir als *"verheiratet".*

Solche verheirateten Kantenpaare können wir im späteren Verlauf (nach diesem Teilschritt) als Einheit betrachten: Wir tun später dann so, als ob es jeweils nur einen einzigen Kantenstein gäbe und versuchen, den Würfel nach der ganz normalen 3x3x3-Methode zu lösen.

Schon einmal vorab: Vermutlich wird uns dies nicht ohne weiteres gelingen. Es gibt nämlich eine gewisse Wahrscheinlichkeit dafür, dass in den Folgeschritten *OLL oder PLL* verschiedene Spezialfälle auftreten, die uns ein bisschen zurückwerfen, weil sie uns Zusatzprobleme bescheren: Das sind sie so genannten Parity-Cases. Diese Paritys kann es beim normalen 3x3x3 Würfel konstruktionsbedingt nicht geben. Diese Spezialfälle schauen wir uns dann gesondert an und beseitigen sie, erst dann können wir weitermachen. Wenn wir so vorgehen können wir im Anschluss daran den 4x4x4 Würfel wie einen 3x3x3 Würfel betrachten und entsprechend lösen.

Doch zunächst: Zur Verheiratung unverbundener Kantenpaare:

Ich werde nun zunächst allgemein beschreiben, nach welcher *Idee* wir vorgehen, um das Ziel "Zwölf verheiratete Kantenpärchen" zu erreichen.

Man kann sich das Vorgehen wie einen Programmablauf vorstellen: In einer äußeren Schleife gehen wir nacheinander alle 12 unverheirateten Kantenpärchen durch. Mit welchem Kantenpaar man beginnt und in welcher Reihenfolge man vorgeht ist egal — es kommen letzten Endes alle Kantenpaare dran. In der inneren Schleife findet dann jeweils die **Verheiratung** statt: Wir stellen sie passend nebeneinander. Damit die Kantenpärchen durch nachfolgende Verheiratungen anderer Kantenpärchen nicht sofort wieder (bildlich gesprochen) "die Scheidung" einreichen, lassen wir sie **"auf Hochzeitsreise"** gehen (wir sichern sie an einen anderen Ort am Würfel weg.)

Das Endziel ist, dass alle Kantenpaare am Würfel nebeneinander stehen ("verheiratet sind").

Wichtig ist, dass wir jedes gelöste Kantenpärchen im Kopf mitzählen, was mit ein bisschen Übung (wenn der Verheiratungsalgorithmus quasi ohne großes Nachdenken wie von selbst abläuft — *muscle memory*) keine Schwierigkeiten macht. Warum? Weil wir nur 9 Kantenpärchen stur und ohne großes Nachdenken verheiraten können. **Bei den letzten 3 Kantenpärchen müssen wir aufpassen** und überprüfen, dass wir nicht in den Fall hineinlaufen, dass immer genau 2 Kantenpärchen unverheiratet sind (und das Verheiraten zur Scheidung eines anderen, dritten Kantenpärchens führt. — Dazu weiter unten mehr.)

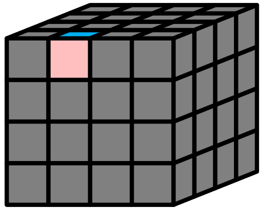
Grundsatz:

Wir vereiraten die Kanten immer an der Frontseite im Top Layer und sichern sie dann seitlich weg.

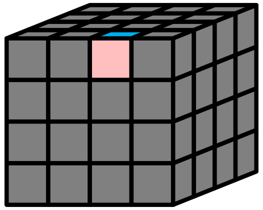
Um das zu erreichen, bringen wir auf der Frontalseite des Würfels je einen Kantenstein ins Top-Layer und seinen Co-Kantenstien in den Bottom-Layer.

Dabei dürfen wir weder Slicedrehungen machen, noch Broad-Moves anwenden, denn diese würden die inneren Layers betreffen und die Ordnung unserer schon perfekt vereinten Mittelsteine zerstören — man braucht aber auch keine Broad- oder Slice-Moves, es genügen die vorhandenen normalen R-, L-, F-, B-Drehungen sowie Drehungen des Deckels und des Boden des Zauberwürfels...

Damit sollten sich folgende beiden Möglichkeiten ergeben, wo sich der Kantenstein vor dem Verheiraten im Top-Layer auf der Frontalebene befinden kann:



oder



(Da im obigen Bild nur ganz allgemein die Positionen gezeigt werden sollen, wurden die abgebildeten Kantensteine auf dem obigen Bild mit nicht standardkonformen Farben dargestellt.)

Sobald wir das bewerkstelligt haben, versuchen wir den Co-Kantenstein in den Botton-Layer zu bekommen.

Damit wir aber die Kantensteine verheiraten können, *darf der Co-Kantenstein nicht direkt unter seinem Partnerstein stehen, sondern er muss im* ***benachbarten Layer*** *sein.*

Unsere Absicht ist also, dass wir durch einen *Broad Move* die Kantensteine farbgleich nebeneinander stellen können (oder in Notation gesprochen: r↑ bzw. $\uparrow l$ wird ein Teil unseres Verheiratungsalgorithmus werden...)

Damit ergeben sich zwei Möglichkeiten: Entweder steht das Kantensteinpaar schon versetzt, dann können wir direkt mit dem Verheiratungsalgorithmus weitermachen. Falls das Kantensteinpaar aber noch direkt untereinander steht, müssen wir uns die gewünschte Ausgangslage zur Verheiratung erst einmal erarbeiten. Das können wir entweder völlig intuitiv tun oder mit einem Algorithmus:

Möglichkeit 1: Rein intuitives Vorgehen:
Wir drehen den Deckelboden um eine Vierteldrehung so herum, dass der Kantenstein am Würfelboden zur weiter von ihm entfernen Würfelseite hin wandert - also der am Bottom Layer in der Mitte links stehende Kantenstein zur rechten Würfelseite und der in der Mitte rechts stehende Kantenstein zu linken Wüfelseite hin wandert:

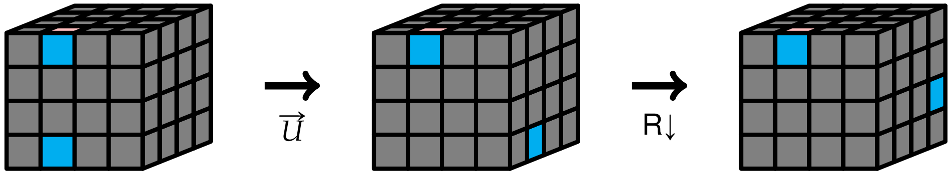
Danach drehen wir den Kantenstein weiter an die Würfelrückseite, indem wir eine Vierteldrehung der entsprechenden Würfelseite nach vorne vornehmen.

Je nach Fall muss man dann also eine der beiden Würfelseiten herunterdrehen: Entweder R↓ oder im anderen Fall: $\downarrow L$.

Dadurch dass die Würfelseite von uns nach vorne heruntergezogen wird, wandert der Kantenstein ganz automatisch an die Würfelrückseite.

Zum besseren Verständnis wollen wir diesen Vorgang einmal für den linksseitigen Fall skizzieren (der rechtsseitige Fall wird dann analog aber spiegelverkehrt ausgeführt.)

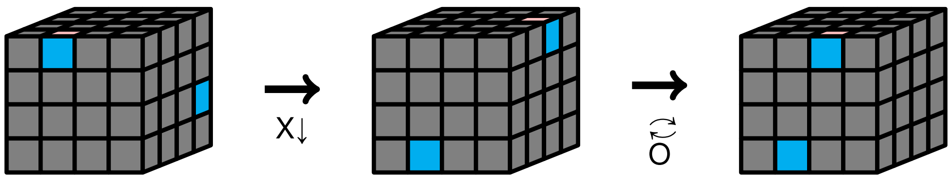
Beispiel:
Den am Boden eher links stehenden Kantenstein für Verheiratung vorbereiten:



(Da im obigen Bild nur ganz allgemein die Positionen gezeigt werden sollen, wurden die abgebildeten Kantensteine auf dem obigen Bild mit nicht standardkonformen Farben dargestellt.)

Danach kippen wir mit dem Move X↓ den Würfel (global) nach vorne.

Unsere Würfelrückseite ist dann die neue obere Seite und die bisherige Würfeloberseite ist nun am Würfelboden. Der bisherige Co-Kantenstein steht nun oben im Deckel an der Seite. Also drehen wir den Deckel in die Frontalebene. Im Ergebnis soll das dann so aussehen, dass die Kantenpaare benachbart in den beiden inneren Layers liegen (einer oben, einer unten). Veranschaulichen wir dies noch einmal, indem wir den letzten Status der obigen Drehsequenz aufgreifen und fortfahren:

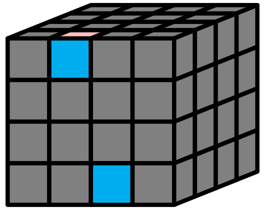


(Da im obigen Bild nur ganz allgemein die Positionen gezeigt werden sollen, wurden die abgebildeten Kantensteine auf dem obigen Bild mit nicht standardkonformen Farben dargestellt.)

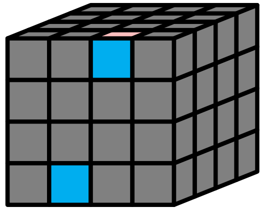
Möglichkeit 2: Algorithmus anwenden:
Egal, ob sich das Kantensteinpärchen auf dem 2. oder 3. horizontalen Slice befindet, können wir folgenden Algorithmus anwenden:

$\overrightarrow{F}$ $R\uparrow$ $\overleftarrow{F}$ $\overset{\circlearrowleft}{O}$

Im Ergebnis sollten wir dann fallabhängig das jeweilige Kantensteinpärchen verheiratungsbereit bekommen:



oder



Die "Verheiratung" von Kantensteinpärchen:
Mit einem nach oben gerichteten *"broad move"* (r↑ Drehung oder einer ↑*l*-Drehung) stellen wir die beiden Kantensteine passend nebeneinander, und **verheiraten** sie (bildlich gesprochen) auf diese Weise. Ab sofort können wir sie damit als eine *gemeinsame Einheit* betrachten.

→ Die Folge davon? - Augenblicklich zerstören wir dabei unsere schön ausgerichteten Mittelsteine, weshalb wir den Broad-Move im weiteren Verlauf wieder rückgängig machen müssen. Zuvor sichern wir aber durch Deckeldrehung das neu verheiratete Kantenpaar weg. — Wir "schicken das verheiratete Pärchen sozusagen auf Hochzeitsreise"...)

Entsprechend haben wir 2 Algorithmen zum Verheiraten von Kantenpärchen:

Rechts-seitiges Verheiraten: *Der rechte Kantenstein wandert hoch*

Beim rechts-seitigen Verheiraten müssen wir immer darauf achten, dass im Top-Layer auf der **rechten Seite** kein schon verheiratetes Kantenpaar steht! Denn dieses würden wir nämlich durch den nachfolgenden Algorithmus auftrennen ("wieder scheiden"). Sollte also dort eines stehen, müssen wir *vorab* die rechte Würfelseite so lange drehen, bis sich dort ein unverheiratetes Kantenpärchen befindet.

$$\mathrm{r}\uparrow \quad \overset{\circlearrowleft}{\mathrm{O}} \quad \mathrm{R}\downarrow \quad \overset{\circlearrowright}{\mathrm{O}} \quad \mathrm{r}\downarrow$$

Links-seitiges Verheiraten: *Der linke Kantenstein wandert hoch*

Auch beim links-seitigen Verheiraten müssen wir immer darauf achten, dass im Top-Layer auf der **linken Seite** kein verheiratetes Kantenpaar steht. Denn auch hier würden wir das bereits verheiratete Kantenpaar wieder trennen ("scheiden"). Sollte also dort eines stehen, müssen wir *vorab* die linke Würfelseite so lange drehen, bis sich dort ein unverheiratetes Kantenpärchen befindet.

$$\uparrow l \quad \overset{\circlearrowright}{\mathrm{O}} \quad \downarrow L \quad \overset{\circlearrowleft}{\mathrm{O}} \quad \downarrow l$$

Nochmals: *Broad Moves* in Gestalt von r-Moves, l-Moves, u-Moves, o-Moves und b-Moves dürfen nur als einführende Setup-Moves benutzt werden, gefolgt von unkritischen Randdrehungen und wieder begradigend durch eine Zurücknahme der zerstörerischen Wirkung via Setback-Moves, dergestalt, dass der Setback-Move in die entgegengesetzte Richtung, die der Setup-Move hatte, erfolgt. Auf diese Weise bleiben die Mittelsteinordnungen erhalten.

Im Ergebnis versuchen wir nun auf diese Weise, weitere Kantenpaare zu verheiraten und zwar so lange, bis nur noch 3 der 12 vorhandenen Kantenpaare noch nicht verheiratet sind.

Nur noch 2 unverheiratete Kantenpaare vorhanden?
In diesem Fall sind wir übers Ziel hinausgeschossen, denn wir werden den Würfel nicht lösen können, weil wir bei zwei verbliebenen unverheirateten Kantenpärchen immer ein schon verheirates Pärchen scheiden müssten, um ein anderes zu verheiraten.

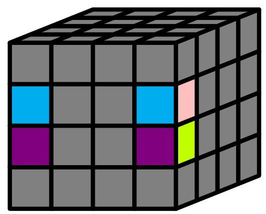
Wir müssen also unsere Vorgehensweise ändern.

Unser Ausweg aus dem Dilemma besteht darin, dass wir im Falle von nur noch 2 unverheirateten Kantenpaaren, zuerst einmal das tun, was wir bei der bisherigen Vorgehensweise genau nicht wollten: *Wir stellen die Kanten so untereinander, dass sie sich in einem gemeinsamen vertikalen Slice befinden, d.h. eine Kante eines Kantenpärchens ist dann im Top-Layer, die andere genau darunter — im gleichen Slice — im Bottom-Layer*.

Den Algorithmus hierfür haben wir oben schon kennengelernt. Da wir die Kantenpärchen aber hier nicht im Top- und Bottomlayer brauchen können, sondern rechts und links, schieben wir zum Abschluss noch einen weiteren Move nach, nämlich eine Vierteldrehung der Frontseite nach rechts:

$$\overrightarrow{F} \quad R\uparrow \quad \overleftarrow{F} \quad \overset{\circlearrowright}{\mathrm{O}} \quad + \quad \overrightarrow{F}$$

Damit befinden sich die Partner jedes der Kantenpaare im gleichen Equator-Slice bzw. stehen sich die Partner jedes der beiden Kantenpaare direkt gegenüber:



Nun können wir einen zweiten Algorithmus anwenden, der dafür sorgt, dass die beiden Kantenpaare ebenfalls verheiratet werden:

Deadlock-Auflösungsalgorithmus Variante 1

$$\overrightarrow{u} \quad R\uparrow \quad \overleftarrow{F} \quad \overset{\circlearrowright}{\mathrm{O}} \quad R\downarrow \quad \overrightarrow{F} \quad \overleftarrow{u}$$

Deadlock-Auflösungsalgorithmus Variante 2

Zugegeben: Ich habe obigen Algorithmus in der freien Wildbahn des Internet gefunden — und er funktioniert einwandfrei und tut genau das, was er soll. Jedoch fand ich den Algorithmus recht ungeeignet, um Fingertricks auszuführen bzw. auch *"rumpelig"*, was das *Muscle Memory* betrifft. Daher habe ich mir die Freiheit genommen, den Algorithmus *auf die linke Seite zu transponieren*, einfach, um nicht so viele Re-Grips zu haben. Er gefällt mir besser – im Endeffekt passiert jedoch am Würfel genau dasselbe wie zuvor, nur aus einer anderen Perspektive und mit *"übersetzten Moves"*:

$$\overset{\circlearrowright}{\mathrm{o}} \quad \overrightarrow{F} \quad \uparrow L \quad \overset{\circlearrowright}{\mathrm{O}} \quad \overleftarrow{F} \quad \downarrow L \quad \overset{\circlearrowleft}{\mathrm{o}}$$

Eselsbrücke / "Zauberspruch": **Vorwärts-"oFLO", dann Rückwärts-"FLo"**

Fingertricks:

Grundhaltung: Linker Daumen ist an der Front im zweituntersten Layer, linker Zeigefinger liegt hinten links an, so dass sowohl Top-Layer als drittes Layer von unten gleichzeitig bewegt werden kann. Weiterhin: Der rechter Daumen ist auf der Würfelunterseite an der Frontebene.

Dann: Beim ersten Move drückt der linke Zeigefinger hinten die beiden obersten Layers nach rechts (an der Frontseite sieht das wie eine Drehung nach links aus). — Achte dabei darauf, dass der rechte Daumen diese Layers nicht blockiert! Beim zweiten Move zieht der rechte Zeigefinger die Frontseite nach rechts herunter. Im dritten Move ist ein Re-Grip erforderlich. Danach ziehen der linke Daumen und Zeigefinger die linke Seite nach hinten. Beim vierten Move zieht der rechte kleine Finger die Oberseite nach rechts (von vorne sieht das wie eine Bewegung nach links aus). Im fünften Move drückt der rechte Daumen die Frontseite nach links (entgegeben dem Uhrzeitgersinn). Dann zieht die linke Hand die linke Würfelseite wieder nach vorne herunter. Beim letzten Move zieht der linke Zeigefinger die beiden obersten Ebenen nach vorne (ergibt eine Linksdrehung).

→ Egal, ob man Variante 1 oder 2 anwendet: Im Ergebnis sind damit alle Kantenpärchen verheiratet und wir können mit dem nächsten Schritt "Weißes Kreuz am Würfelboden bilden" (siehe entsprechendes Kapitel) weitermachen.

Das Verheiraten der 3 letzten Kantenpaare:
Haben wir noch 3 unverheiratete Kantenpaare, müssen wir unser Vorgehen etwas anpassen, damit wir tatsächlich alle Pärchen verheiratet bekommen.

Wie gehabt, versuchen wir, auf der Frontalseite ein Kantenpärchen ins Top- und Bottom-Layer bekommen. Wenn sie direkt untereinander stehen, dann wenden wir den gezeigten Weg an, um sie in verschiedene innere vertikale Layers zu bekommengehen. Wenn sie schon versetzt stehen (wie zuvor gezeigt) und wir zudem ein unverheiratetes Pärchen auf die passende Seite des Top-Layers gestellt haben, könnten die Kantensteine (rein theoretisch) verheiratet werden — was wir aber erst mal **unterlassen.**

Statt dessen machen wir etwas anderes:

Check: *Prüfe: Lassen sich alle Kantenpaare verheiraten?*
Die Prüfung geht ziemlich einfach. Zunächst prüfen wir, ob sich durch Hochdrehen via r↑ oder $\uparrow l$ ein frontal im Bottom-Layer stehender Kantenstein mit seinem Co-Kantenstein verheiraten lässt. (Das wird in der Regel der Fall sein, wir haben ja die Steine so hingedreht). Aber wir müssen auch prüfen, ob sich die anderen Kantensteine problemlos verheiraten lassen: Dazu drehen wir den Deckel so, dass das nicht verheiratete, im Top-Layer seitlich stehende Paar an die Frontalseite kommt. Und dann muss sich das andere Kantensteinpaar durch Hochdrehen via r↑ oder $\uparrow l$ ebenfalls verheiraten lassen können (sprich: gleiche Farben aufweisen).

Ist dies der Fall, ist alles gut... ...und wir können den Deckel zurückdrehen und wissen, dass sich alle Kantensteinpaare ohne Probleme verheiraten lassen werden. Und das tun wir dann auch und können danach direkt zum nächsten Schritt: *"Weißes Kreuz bauen!"* weitergehen.

Falls der Check jedoch zeigt, dass das nicht möglich ist, dann muss man einen Algorithmus anwenden, der dafür sorgt, dass die beiden Kantensteine, die im Top-Layer auf der Würfelseite stehen, **ihre Plätze miteinander tauschen.** — Sie bleiben zwar als unverheirates Pärchen beieinander, aber sie tauschen ihre Position am Würfel miteinander.

Der Platzvertauschungs-Algorithmus geht wie folgt:

Man zieht diejenige Würfelseite nach vorne herunter, bei der die Kanten getauscht werden sollen. Dann schiebt man mit einer F-Drehung die Kanten wieder nach oben ins Top-Layer und dreht am Deckel, so dass die Kanten wieder im Top-Layer an ihren alten Platz auf der Seite wandern - nur eben vertauscht. Danach nimmt man die F-Drehung durch die entsprechende Gegendrehung wieder zurück.

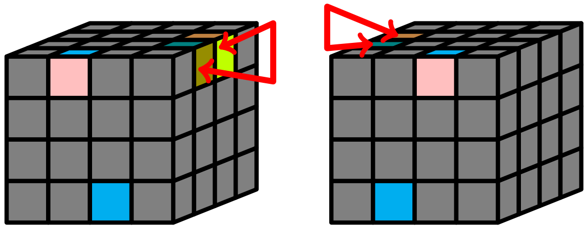
Rechtsseitige Pärchen sollen Plätze miteinander tauschen:

$$R\downarrow \quad \overleftarrow{F} \quad \overset{\leftrightarrows}{\mathrm{O}} \quad \overrightarrow{F}$$

Linksseitige Pärchen sollen Plätze miteinander tauschen:

$$\downarrow L \quad \overrightarrow{F} \quad \overset{\rightleftarrows}{\mathrm{O}} \quad \overleftarrow{F}$$

Im Ergebnis werden hier die beiden Kantensteine vertauscht:



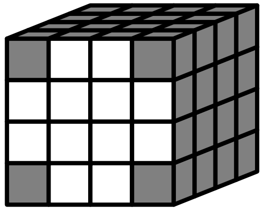
Nun prüfen wir erneut und es sollte sich zeigen, dass sowohl das normale Verheiraten klappt, als auch das Verheiraten, wenn wir am Deckel das unverheiratete Pärchen auf die Frontseite hineindrehen und die Heirat überprüfen: Das sollte möglich sein. (Danach drehen wir den Deckel zurück). Nun sollten wir alle Kantenpaare verheiraten können — und das tun wir dann auch!

Fazit:
Hier sind jetzt alle Kantenpaare passend verheiratet.

→ Wir können nun fortfahren und das weiße Kreuz bilden.

Bilden des weißen Kreuzes am Würfelboden:

Wir betrachten ab jetzt die verheirateten Kantenpaare als einen einzigen Kantenstein. Wie bei der normalen Anfängerlösung des 3x3x3 bringen wir nun alle weißen Kantensteine nach unten auf den Würfelboden, um das so genannte "weiße Kreuz" hinzubekommen. Dabei achten wir darauf, dass die Seitenfarben der weißen Kantensteine beim Einsetzen am Boden korrket zur Seitenfarbe des Würfels passen. Der Würfelboden sieht am Ende (von unten betrachtet) wie folgt aus (wir haben, damit wir etwas sehen können, hier den Würfel seitlich gekippt - die Frontseite entspricht also dem Würfelboden.)



Sicht auf den Würfelboden: Es kann sein, dass zufällig am Würfelboden noch weitere weiße Steine in den hier grau dargestellten Ecken vorhanden sind. Aber das weiße Kreuz muss in jedem Fall auch vorhanden sein.

Weiße Ecksteine ausrichten/lösen → Weiße Grundseite fertig stellen:

Nun stellen wir die Ecksteine nacheinander in das Top-Layer, und zwar über ihren jeweiligen, am Würfelboden befindlichen Einfügeslot. Mit mehrfachem so genannten *"Sexy Move"* ($R\uparrow$ $\overset{\curvearrowleft}{O}$ $R\downarrow$ $\overset{\curvearrowright}{O}$) locken wir alle 4 weißen Ecksteine an ihrer Homeposition ein und stellen damit die weiße Grundseite bzw. den Würfelboden fertig.

Lösen der drei unteren Layers → Erreiche Status *F3L*:

Da wir die Kantenpärchen weiter jeweils als "einen einzigen Kantenstein" behandeln, können wir wie beim normalen 3x3x3 vorgehen und die Kantenpärchen nach der "Beginners Method" entweder links oder rechts einsetzen. Der Unterschied ist eigentlich nur, dass wir damit nicht nur wie beim 3x3x3 den Status *F2L* erreichen, sondern sofort und automatisch gleich drei Ebenen gelöst haben, nämlich: *F3L*.

Vorbemerkungen zum OLL, insbesondere: *Vermeidbare Fehler*

Wir versuchen beim OLL, die gelbe Deckelseite zu lösen, und zwar ganz wie beim 3x3x3. Mit anderen Worten: Wir versuchen, die Lösungskaskade *"Gelbe Linie"* → *"Gelbes Kreuz"* → *"Gelber Fisch"* → *"Gelbe Oberseite"* hinzubekommen.

Dabei können wir einen Fehler vermeiden, der uns weit zurückwerfen wird: Die aus dem 3x3x3 bekannten *broad Moves* dürfen wir auf keinen Fall benutzen! — Sie zerstören die verheirateten Kantenpaare und scheiden sie, so dass sie nicht mehr direkt nebeneinander stehen! Beispielsweise dürfen wir nicht *"einfach mal so, wie beim 3x3x3"* die ersten beiden frontalen Layers gemeinsam drehen, um die lange OLL-Kaskade abzukürzen.

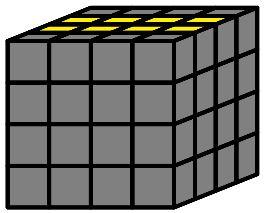
Und: Wir sollen uns nicht irritieren lassen: Wir versuchen ganz stur erst einmal unsere gelbe doppelte Linie hinzubekommen. Das war ja in der Singmaster-Notation der FRU-R'U'F' aus der Beginners Method:

$\overrightarrow{F}$ R↑ $\overset{\curvearrowleft}{O}$ R↓ $\overset{\curvearrowright}{O}$ $\overleftarrow{F}$

(Als Eselsbrücke murmle ich dabei: *"Freundliche Rechtsdreher operieren rückwärts ohne Fehler!"*)

(Manche sagen dazu auch "Balken-Move", obwohl es ein Algorithmus und kein einzelner Move ist...)

Wenn wir mindestens die doppelte gelbe Linie hinbekommen haben (es können aber auch noch mehr gelbe Sticker nach oben zeigen), ist dieser Teilschritt erledigt:



Anschließend wollen wir ein gelbes Kreuz hinbekommen. Hierzu machen wir wieder den obigen Algorithmus, achten dabei aber darauf, dass die gelbe Linie (wie im Bild oben gezeigt) von links nach rechts verläuft (und nicht etwa: von vorne nach hinten!) — gelingt uns dies, dann haben wir ein gelbes Kreuz. Von diesem *"State"* aus können wir ganz normal die gelbe Würfeloberseite fertigstellen.

Wir müssen uns nur, wie schon gesagt davor hüten, dass wir keine unbedarften Broad-Moves machen, damit wir die Mittelsteinordnung nicht zerstören!

Wenn wir jedoch nach mehrmaliger Anwendung des Sune-Algorithmus (siehe meine Anleitung zur Beginners Method) kein gelbes Kreuz hinbekommen, sondern ein anderes Muster auftaucht, z. B. ein dickes gelbes ”T-”Zeichen, ein ”P-”Zeichen, ein ”U-”Zeichen oder ein ”Y-”Zeichen, haben wir einen so genannten OLL Parity Case!

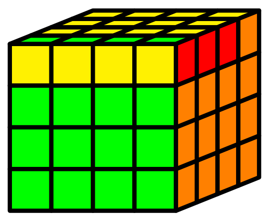
Prüfen: Haben wir einen OLL-Parity Case?
Betrachten wir den Würfel: Ein kurzer Blick verrät uns, ob es eine ungerade Anzahl an Kantenpärchen gibt, bei der die gelbe Seite nicht nach oben zeigt, sondern zur Seite (also bei einem Kantenpärchen oder bei drei Kantenpärchen.) Wenn wir das entdecken, brauchen wir erst gar nicht zu versuchen, das gelbe Kreuz irgendwie hinzubekommen, denn das wird nicht möglich sein — es handelt sich um seinen so genannten ***OLL-Parity-Case.*** Diesen müssen wir zuerst einmal durch einen **OLL-Korrekturalgorithmus** beseitigen:

OLL-Parity-Korrekturalgorithmus:

$(r\uparrow\ \overset{2\circlearrowleft}{\mathrm{O}})\ X\uparrow\ 2\bullet(r\uparrow\ \overset{2\circlearrowleft}{\mathrm{O}})\ (r\downarrow\ \overset{2\circlearrowleft}{\mathrm{O}})\ (\downarrow\ell\ \overset{2\circlearrowleft}{\mathrm{O}})\ (r\downarrow\ \overset{2\circlearrowleft}{\mathrm{O}})\ (r\uparrow\ \overset{2\circlearrowleft}{\mathrm{O}})\ (r\downarrow\ \overset{2\circlearrowleft}{\mathrm{O}})\ r\downarrow$

Tipps:
1. Bei X↑ kippen wir den Würfel global nach hinten.
2. Die Deckeldrehungen sind immer 180-Grad-Turns (2 Quarter-Turns) nach rechts.

Was bewirkt dieser Algorithmus?
Er flipped alle 4 Cubies, die im Top-Layer frontal stehen. Und ganz wichtig: Er verändert **keine** Würfelchen (Cubies) mehr, die sich unterhalb des Deckels in den Layern 1-3 befinden! — Er macht aber außerdem auch noch etwas anderes: Er vertauscht die seitlich im Top-Layer stehenden linken und rechten Cubies, aber mit Ausnahme der beiden Back-Cubies der hinteren Ecken:



Eselsbrücke für den Algorithmus:
Ich habe mir lange überlegt, wie man sich den gerade dargestellten OLL-Parity-Korrekturalgorithus am besten merken kann. Und ich bin darauf gekommen, dass ich mir das am besten als kleines, frei erfundenes Lied merke. (Die Melodie mag sich jeder selbst überlegen). Die Liedzeilen enden oft auf '"zwei-drei!", was besagt, dass man den Decke mit dem Zeigefinger und Ringfinger, die an der Deckelrückseite liegen (Fingertricks!) zu sich herziehen soll. Das Lied geht so:

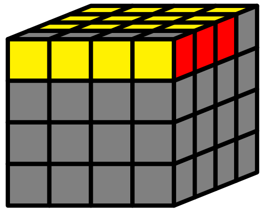
Das Lied des OLL-Korrekturalgorithmus:

Hey, es geht jetzt hoch, zwei-drei!
Und jetzt kippt die Welt nach hinten!
dann geht es hoch, zwei-drei!
dann geht es hoch, zwei-drei!
dann geht es runter, zwei-drei!
dann geht es ***links*** *herunter, zwei-drei!*
dann geht es runter, zwei-drei!
dann geht es hoch, zwei-drei!
dann geht es runter, zwei-drei!
dann geht es nochmal runter!

Im Nachfolgenden werden verschiedene Muster gezeigt, die einem OLL-Parity entsprechen und wie man diese Parity-Cases auflöst.

Fall: Drei gelbe Viererbars
Das ist exakt der Fall, den wir oben, als wir die Wirkungsweise des OLL-Parity-Algorithmus beschrieben haben, herauskommt, wenn man den Algorithmus auf einen bereits gelösten Würfel anwendet.

Demzufolge:



Drei gelbe Viererbars anstelle eines

gelben Kreuzes? → OLL-Parity-Case

OLL-Parity-Korrekturalgorithmus:

Tipps: Die Deckeldrehung ist immer zweimal nach rechts (180 Grad), die Broad-Moves dagegen immer Vierteldrehungen (90 Grad). Und: Zu Beginn halten wir den Cube so, dass die seitlichen Gelbstrahler an der Front stehen.

$(r\uparrow\ {}^{2}\overset{\curvearrowleft}{O})\ X\uparrow\ 2\bullet(r\uparrow\ {}^{2}\overset{\curvearrowleft}{O})\ (r\downarrow\ {}^{2}\overset{\curvearrowleft}{O})\ (\downarrow\ell\ {}^{2}\overset{\curvearrowleft}{O})\ (r\downarrow\ {}^{2}\overset{\curvearrowleft}{O})\ (r\uparrow\ {}^{2}\overset{\curvearrowleft}{O})\ (r\downarrow\ {}^{2}\overset{\curvearrowleft}{O})\ r\downarrow$

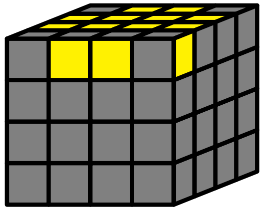
Tipps:
1. Bei X↑ kippen wir den Würfel global nach hinten.
2. Die Deckeldrehungen sind immer 180-Grad-Turns (2 Quarter-Turns) nach rechts.

Was bringt uns der Algorithmus?

Er macht, dass die vier falsch geflippte Kantensteine an der Würfelvorderseite im Top-Layer nun richtig herum geflipped sind. Zusätzlich vertauscht er die seitlichen Cubies. Damit ist die gelbe Oberseite bereits gelöst und wir haben den Status OLL erreicht.

Ergebnis: Status OLL erreicht.

Fall: T-Shape (Variante 1: Zwei gelbe Front- und Seitenstrahler)



Auf dem Kopf stehender, gelber T-Shape anstelle eines gelben Kreuzes → OLL-Parity-Case

OLL-Parity-Korrekturalgorithmus:

Tipps: Die Deckeldrehung ist immer zweimal nach rechts (180 Grad), die Broad-Moves dagegen immer Vierteldrehungen (90 Grad). Und: Zu Beginn halten wir den Cube so, dass die seitlichen Gelbstrahler an der Front stehen.

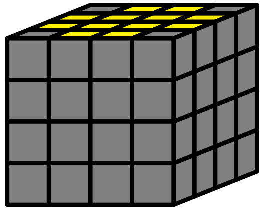
(r↑ ^{2}O) X↑ 2 •(r↑ ^{2}O) (r↓ ^{2}O) (↓ℓ ^{2}O) (r↓ ^{2}O) (r↑ ^{2}O) (r↓ ^{2}O) r↓

Tipp: Bei X↑ kippen wir den Würfel global nach hinten.

Was bringt uns der Algorithmus?

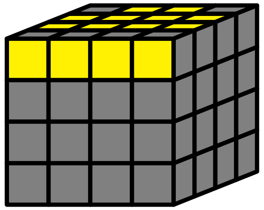
Er macht, dass das verheiratete, falsch geflippte Kantenpärchen nun richtig herum geflipped ist. Die Back-Corners werden vom Algorithmus nicht angetastet und bleiben weiter ungelöst. Die frontalen Corners werden geflipped, aber da es sich um einen T-Shape und nicht um einen Y-Shape handelt, bleiben die beiden frontalen Corners ungelöst. Wenn wir also obige Ausgangslage voraussetzen, haben wir damit das gelbe Kreuz gelöst, denn die Ecken sind ja noch ungelöst gewesen. Nun können wir in der Folge die gelbe Oberseite lösen und den Status OLL erreichen.

Ergebnis: Gelbes Kreuz sichtbar — plus weitere gelbe Sticker:



Von hier aus können wir dann wie dann ganz wie beim normalen 3x3x3 weitermachen: entweder können wir den Würfel insgesamt lösen oder es treten andere Parity-Fälle auf (mehr dazu weiter unten).

Fall: T-Shape (Variante 2: Vier gelbe Frontstrahler)



Auf dem Kopf stehender, gelber T-Shape anstelle eines gelben Kreuzes → OLL-Parity-Case

OLL-Parity-Korrekturalgorithmus:

Tipps: Die Deckeldrehung ist immer zweimal nach rechts (180 Grad), die Broad-Moves dagegen immer Vierteldrehungen (90 Grad). Und: Zu Beginn halten wir den Cube so, dass die seitlichen Gelbstrahler an der Front stehen und sozusagen gelbe Frontstrahler bilden.

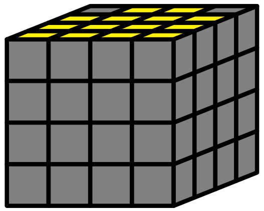
(r↑ ^{2}O) X↑ 2 •(r↑ ^{2}O) (r↓ ^{2}O) (↓ℓ ^{2}O) (r↓ ^{2}O) (r↑ ^{2}O) (r↓ ^{2}O) r↓

Tipp: Bei X↑ kippen wir den Würfel global nach hinten.

Was bringt uns der Algorithmus?

Er macht, dass alle 4 zur Frontseite herausstrahlenden gelben Stickers nach oben zeigen und somit richtig herum geflipped sind. Wenn wir also obige Ausgangslage voraussetzen, haben wir damit das gelbe Kreuz gelöst, denn zwei Ecken auf der Rückseite sind ja vor Anwendung des obigen Algorithmus noch ungelöst gewesen (und sie sind es auch jetzt immer noch, denn der Algorithmus fasst sie nicht an.) Nun können wir weiter die gelbe Oberseite lösen und den Status OLL erreichen.

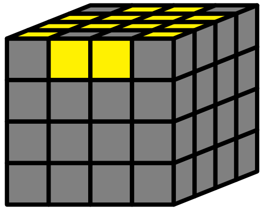
Ergebnis: Gelbes Kreuz sichtbar — plus weitere gelbe Sticker:



Von hier aus können wir dann wie dann ganz wie beim normalen 3x3x3 weitermachen, um die gelbe Oberseite zu lösen und den Status OLL zu erreichen.

Analoger Fall: Y-Shape

Der Y-Shape liegt vor, wenn das gelbe Muster dem Y-Zeichen entspricht, d.h. zwei Corners zeigen mit ihrer gelben Seite nach oben und zwei Corners zeigen mit einer *anderen Farbe* nach oben. Es gibt eine gelbe Querlinie und zwei verheiratete Eckpaare, die nicht mit der gelben Seite nach oben zeigen.



Auf dem Kopf stehender, gelber Y-Shape anstelle eines gelben Kreuzes → OLL-Parity-Case

OLL-Parity-Korrekturalgorithmus:

Tipps: Die Deckeldrehung ist immer zweimal nach rechts (180 Grad), die Broad-Moves dagegen immer Vierteldrehungen (90 Grad). Und: Zu Beginn halten wir den Cube so, dass die seitlichen Gelbstrahler an der Front stehen.

(r↑ $\overset{2\rightleftarrows}{\mathrm{O}}$) X↑ 2 •(r↑ $\overset{2\rightleftarrows}{\mathrm{O}}$) (r↓ $\overset{2\rightleftarrows}{\mathrm{O}}$) (↓ℓ $\overset{2\rightleftarrows}{\mathrm{O}}$) (r↓ $\overset{2\rightleftarrows}{\mathrm{O}}$) (r↑ $\overset{2\rightleftarrows}{\mathrm{O}}$) (r↓ $\overset{2\rightleftarrows}{\mathrm{O}}$) r↓

Tipp: Bei X↑ kippen wir den Würfel global nach hinten.

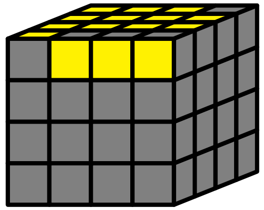
Was bringt uns der Algorithmus?

Er macht, dass das verheiratete, falsch geflippte Kantenpärchen nun richtig herum geflipped ist. Aber die schon richtig geflippten frontalen Corners werden dabei falsch herum geflipped (was nicht tragisch ist — wir wollen ja nur den Parity Case beseitigen!) Und: Die beiden Backside-Corners, die ja auch noch nicht richtig herum geflipped waren, bleiben weiter falsch herum geflipped, da der Algorithmus diese Corners nicht antastet. Wenn wir also obige Ausgangslage voraussetzen, haben wir damit das gelbe Kreuz gelöst und können damit fortfahren, die gelbe Oberseite zu lösen, um so den Status OLL zu erreichen.

Ergebnis: Gelbes Kreuz erreicht.
Weiter fortfahren mit dem Lösen der gelben Würfeloberseite.

Analoger Fall: P-Shape

Der P-Shape liegt vor, wenn das gelbe Muster dem P-Zeichen entspricht, d.h. zwei Corners zeigen mit ihrer gelben Seite nach oben und zwei Corners zeigen mit einer anderen Farbe nach oben. Es gibt eine gelbe Querlinie und zwei verheiratete Eckpaare, die zusammen mit einem Corner ein Triplett bilden, das nicht mit der gelben Seite nach oben zeigt.



Auf dem Kopf stehender, gelber P-Shape anstelle eines gelben Kreuzes → OLL-Parity-Case

OLL-Parity-Korrekturalgorithmus:

Tipps: Die Deckeldrehung ist immer zweimal nach rechts (180 Grad), die Broad-Moves dagegen immer Vierteldrehungen (90 Grad). Und: Zu Beginn halten wir den Cube so, dass die seitlichen Gelbstrahler an der Front stehen und sozusagen gelbe Frontstrahler bilden.

(r↑ $\overset{2\rightleftarrows}{\mathrm{O}}$) X↑ 2 •(r↑ $\overset{2\rightleftarrows}{\mathrm{O}}$) (r↓ $\overset{2\rightleftarrows}{\mathrm{O}}$) (↓ℓ $\overset{2\rightleftarrows}{\mathrm{O}}$) (r↓ $\overset{2\rightleftarrows}{\mathrm{O}}$) (r↑ $\overset{2\rightleftarrows}{\mathrm{O}}$) (r↓ $\overset{2\rightleftarrows}{\mathrm{O}}$) r↓

Tipp: Bei X↑ kippen wir den Würfel global nach hinten.

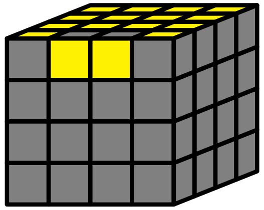
Was bringt uns der Algorithmus?

Er macht, dass das verheiratete, falsch geflippte Kantenpärchen nun richtig herum geflipped ist. Ebenso wird der rechte Top-Layer-Corner richtig herum geflipped sein. Dagegen wird der ursprünglich richtig herum geflippte Corner wieder falsch herum geflipped (was nicht tragisch ist!) Die beiden Corner am Ende bleiben vom Algorithmus unberührt, so dass der rechts hinten stehende Corner weiter falsch herum geflipped bleibt. Wenn wir also obige Ausgangslage voraussetzen, haben wir damit das gelbe Kreuz gelöst und können damit fortfahren, die gelbe Oberseite zu lösen, um so den Status OLL zu erreichen.

Ergebnis: Gelbes Kreuz erreicht.
Weiter fortfahren mit dem Lösen der gelben Würfeloberseite.

Analoger Fall: U-Shape

Auch dieser Fall stellt einen OLL-Parity-Case dar: Hier sehen wir auf der gelben Oberseite ein U-Muster und stellen fest, dass alle 4 Top-Layer Corners an der falschen Position stehen.



Auf dem Kopf stehender, gelber U-Shape anstelle eines gelben Kreuzes → OLL-Parity-Case

Auch hier brauchen wir den Korrekturalgorithmus:

Auch hier halten wir den Würfel so, dass das falsch geflippte, verheiratete Kantenpärchen an der Frontalseite steht und uns die gelben Stickers sozusagen "anstrahlen". Dann:

(r↑ ^{2}O) X↑ 2 •(r↑ ^{2}O) (r↓ ^{2}O) (↓ℓ ^{2}O) (r↓ ^{2}O) (r↑ ^{2}O) (r↓ ^{2}O) r↓

Tipp: Bei X↑ kippen wir den Würfel global nach hinten.

Was bringt uns der Algorithmus?

Er flipped das verheiratete Kantenpärchen richtig herum und flipped die bisher korrekt ausgerichteten frontalen Corners falsch herum (was nicht tragisch ist!) Auf der bisher gelben Oberseite entsteht ein Baum (also drei gelbe Viererbars darunter mittig eine gelbe Zweierbar). Daraus können wir dann durch Standardalgorithmen der Beginners-Methode ein gelbes Kreuz, dann einen gelben Fisch und dann die gelbe Oberseite machen und haben den Zustand OLL erreicht.

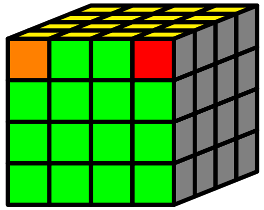
Ergebnis: Gelber Baum erreicht
Weiter fortfahren mit dem Lösen der gelben Würfeloberseite.

PLL-Parity Cases:

Beachte: Ein PLL-Parity-Case kann nur vorliegen wenn die *gesamte Würfeloberseite* bereits gelb ist! - Sind also irgendwelche Steine im Top-Layer falsch geflipped, so kann es sich nicht um einen PLL-Parity-Case handeln, weil wir den Status *"OLL-finished"* noch gar nicht erreicht haben. Ein PLL-Parity-Case kann andererseits auch dann nicht vorliegen, wenn 3 Ecken, bei denen die gelbe Farbe oben ist (OLL-Zustand ist ja erreicht) falsch positioniert sind. Dann kann man den Würfel normal lösen und muss nicht erst überprüfen, ob man einen PLL-Parity-Case hat.

Ansonten gilt: Es gibt für Speedcuber ausgefeilte Algorithmen, mit denen man jeden der vier unterschiedlichen PLL-Fälle optimiert lösen kann. Aber wir wollen die Dinge einfach halten und uns (vorerst) einfach daran erfreuen, dass wir den 4x4x4 überhaupt lösen können :-) Daher verwenden wir einen einfachen Algorithmus, den wir auf jeden der vier PLL-Cases anwenden und der uns den Cube lösbar macht. — Die fortgeschrittenen Methoden haben dann am Ende schon den Cube gelöst, während wir noch die bekannten Algorithmen aus der Anfängerlösung hintendran setzen müssen...

Erster PLL Parity Case: Zwei gegenüber liegende Corners an falscher Position



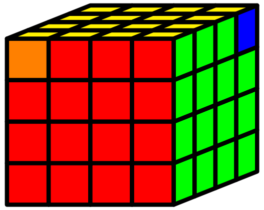
(^{2}o ^{2}r↑) (^{2}O $^{2}R_{III}$ ↑) (^{2}O ^{2}r↑) ^{2}o

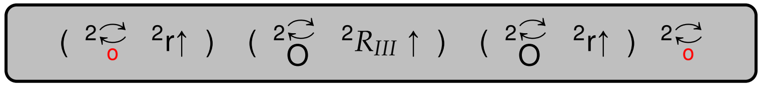
(Der Index III meint, dass das dritte Layer von links als Slice um 180 Grad nach oben zu drehen ist.)

Nun können wir den Würfel nach der normalen 3x3x3 Anfängermethode (oder eben via CFOP-PLL) fertig lösen.

Zweiter PLL Parity Case: Diagonal vertauschte Corners:

Hier halten wir den Würfel erst einmal so, dass die falsch positionierten Corners vorne links und hinten rechts stehen:



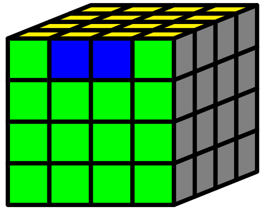


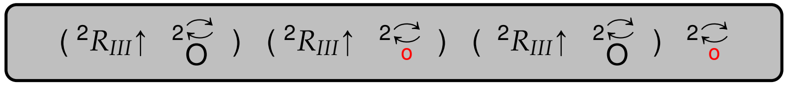
(Der Index III meint, dass das dritte Layer von links als Slice um 180 Grad nach oben zu drehen ist.)

Danach bringen wir wie beim normalen 3x3 zuerst die Kantenpaare des Top-Layers an ihre richtige Stelle bzw. an ihre Home-Position. Anschließend bringen wir wie beim 3x3 die Ecksteine des Top-Layers mit Hilfe des aus der Beginners Methode bekannten "Telefon-Moves" (Ecksteinkarusell) aus. Alternativ können wir auch den A-Perm aus der CFOP-Methode anwenden. Beides mal ist anschließend der 4x4-Cube fertig gelöst.

Dritter PLL Parity Case: Zwei verheiratete Edges gehören auf die Gegenseite und vice versa

Hier halten wir den Würfel erst einmal so, dass die falsch positionierten, verheirateten Edges an der Front und an der Würfelrückseite stehen:





(Der Index III meint, dass das dritte Layer von links als Slice um 180 Grad nach oben zu drehen ist.)

Nun können wir den Würfel nach der normalen 3x3x3 Anfängermethode (oder eben via CFOP-PLL) fertig lösen.

Abschließende Tipps:

In meinen Erklärungen bzw. Anleitung zur Beginners-Methode habe ich in Kapitel 11.7 auch Abkürzungen gezeigt. Hierbei werden oft die ersten beiden frontalen Layers zusammen gedreht. Ähnliche Abkürzungen gibt es für das Muster "Space Invader" usw. Diese kann man auch auf den 4x4-Cube anwenden, man muss dann bei diesen Algorithmen nur folgendes tun: Überall dort, wo man zwei frontale Layers zusammen dreht (f-Drehungen), dreht man jetzt die ersten drei frontalen Layers zusammen. Auf diese Weise kann man einige Schritte hinsichtlich dem Lösen der gelben Seite (des gelben Würfeldeckels) einsparen.

/rhn, 30.12.2025