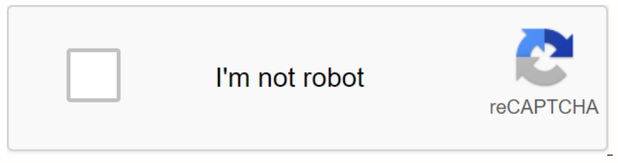

Continue

Roux method pdf

Luvik's cube speed solution was invented by Lu (French pronunciation: [yepu]) Roux is based on block building and corner-first methods. It is notable for its low movement number, lack of rotation, and adaptability to one-handed resolution. Step 1. Build 1x2x3 blocks anywhere in the cube. 2. Build the second 1x2x3 block on the opposite of the first 1x2x3 block, without disturbing the first 1x2x3 block. After this step, there should be two 1x2x3 blocks, one in the lower left and one in the lower right, to move the U and M slices freely. Steps 1 and 2 are called the first two blocks 3. At the same time, the remaining four edges of the top layer (U-slice) are oriented and distorted. You can do this by using CMLL, COLL, or corner OLLs (along with T or J and Y permutations). 4a. Use only M and U moves to orient the remaining six edges (UF, UB, UL, UR, DF, AND DB must be oriented correctly). 4b. Resolve ul and UR edges to maintain edge orientation. After this step, both the left and right layers must be completed. 4c. See the last six edges when you resolve the center and edges of the M slice. Step 3 and 4 are called the last 10 pieces pro, such as the Petrus method, and the Roux method uses less movement than the popular Friedrich method. It is also more intuitive and requires fewer algorithms. After you build the first block, the rotation is removed because the rest of the cube can be mostly resolved by moving R, r, M, and U. CmLL is only 42 cases, and most algorithms are the fast OLLCP of CFOP, so it's one of the best sets of algorithms, and its block building and intuitive properties allow for rapid improvement of the front and inspection, and the LU's LSE steps are very easy to master because they allow for a wider front and fast second-generation MU TPS. Prisoner block buildings can be difficult for beginners to get used to. Reliance on r and M moves can be difficult for some people, so a cupper with a problem with the M turn should probably not use this as the main method (or better M go practice). M-slices are particularly frequently used in the final stages, so if the solver misses a second movie in M2 or the solver misses the last M move, it is more likely to be DNF rather than +2. M is DNF because it uses both The R and L faces as one. M-slices are more difficult with higher-order puzzles. With 7x7x7 and 6x6x6, many people claim that Roux is essentially unavailable. But through practice, you can do well in a large cube. M-slices are very difficult with OH and the OH Roux solver almost always has to take advantage of table abuse, so it cannot be solved with one hand on a table or on another surface. But if there is one practice of m-slices on one hand, it can do very well, but it can only be done with tables (it shouldn't be a problem in competition). Improvement free/nonlinear blocks: You don't need to build the first and second blocks in that order. You can build one part The other one later. This is very useful when there are a lot of free, pre-built blocks and pairs. Mismatched centers: The first two blocks can be built around the wrong center. This increases efficiency and allows the looper to take advantage of pre-built blocks. The center will be modified right before cmll either U M'u'u or u'M'u. Some Roux users have learned different algorithms for each CLL case, each affecting the edge in a different way. This makes lse easier by manipulating the edge direction. This is called CMLLEO. Mismatched blocks: Another improvement is to resolve one block of any of the four possible blocks. This means that the colors of the Two Blocks of The D layer do not have to match. The quickest way to recognize CMLL after building mismatched blocks is to use the NMCLL recognition method. If you plan to use blocks that do not match the new Roux solver, it is best to learn this method of recognizing corners. Otherwise, corner recognition can be difficult if you choose to use a different recognition method. This link has a complete guide to using mismatched blocks. Pinky Pie: A variant proposed by Alex Lau in 2016 that involves using the OLLCP algorithm to orient the UL/UR edges of the D layer while orienting the remaining edges. Users will get 4a skip and 4b steps very easy. This may seem good, but it is the opinion that many quick roosters are simply better not to go to all the hassles that affect the EO stage and place UL/UR on D and recognize OLLCP. There's also a reason why many Roux solvers use this method to use less algorithms and don't want to learn the huge OLLCP eggset. EOLR: A variant proposed by JillU and developed by James Straughan[1]. This variant was used by Alex Lau and used more and documented by Kian Mansour and Iuri Graniro. It combines step 4a and 4b of the method, placing ul+ ur edges at the same time to orient the edges. In eOLRa variants, UL+UR edges are placed on the D layer and EOLRb variants, and the UL+UR edges are resolved. The case is all intuitive and involves performing an EO case from various angles to set up a good arrow. UFUB: Instead of solving the ULUR edge with 4B, UFUB is resolved. This can lead to more efficient solutions, but it makes the front much more difficult. It is most useful for skipping point 4C cases, but requires center recognition and more front. Wrong center: Standard Lou includes orienting all six edges over the center color at the bottom of the block. Instead, we can specify the orientation compared to the front color. As with UFUB, efficiency is improved, but the preceding boundaries are disturbed. This is most useful for known EOLR cases. See also: Advanced techniques for Lulu in different puzzles have a lot to do with a variety of different puzzles inspired by Lu. Select the most well-known method Each puzzle is listed here: 4x4x4 (and other big cubes): Meyer Way, Kenneth's Big Cube Way, Stadler Way, Lewis Way, CR4, Big Roux Square-1: LBL (Square-1) (Cube Shape, Two Corner Edge Corner Blocks, D Edge, PLL), Lean, Screw Skewers: Skrouxb Method Here is the second most notable square-1 method, and they are the second most notable square-1 method behind Vandenbergh and the most popular squares here - the most notable squares here. Notable user external link tutorials/guide example solving this site uses cookies from Google to provide services and analyze traffic. Information about your use of this site is shared with Google. By using this site, you agree to use cookies. Later, common characters are used in a given moving sequence: U, D, L, R, F, B. R. R = Turn on R2 = RR R'= RRR clockwise, and the inner piece is the same piece as the other, and can be moved using one movement: E, M, S. Lowercase should instead set two layers. r = RM' is not perfect, but it is the most widely used. The Roux method is not widely used, but has shown potential through many sub-15 achievements by people like Tom Balough and Jules Manoland. He has also achieved a lot of sub-10 achievements of Austin Moore, Alexander Lau and Kian Mansour. It works with block building in a similar but different way to Petrus. Resolve the 1x2x3 block on the left or right, and then resolve the same block on the other side. CmLL then uses cmll to distort and orient all remaining edge of the last layer. Only M and U-turns are made here. The next step consists of orienting all the remaining bad edges. After the bad edge is oriented, resolve the two edges that fall above the 1x2x3 block at the same time. The last step and the shortest step is to distort the last four edges. Content [Show] Basic Rules EditIng One of the most important things you need to do before you start learning what speedcubing is to know how to solve the Rubik's Cube beginner method. If you know this, you understand that you can't just get all of one color on one side, but you need to put the pieces in the right place as well. Knowing this will help you understand why many algorithms in speed-bbbing methods work. You need to know the color scheme of the cube by mind. To get a quick time, you need to know where the pieces move relative to each other without thinking about it. Notation here is the notation table to be used here: Layered R-right face L-left face U-up face D-down face F-front face B-back face R and L (direction of l) you and D (equator) (direction of the equator) (Equator) (equator) direction) F and B (side) (direction of F) X-Slice F rotation clockwise between the entire cube in the direction of the S-full cube in the direction of the S-whole cube' or i branch turn opposite clockwise direction 2-and-a half rotating w or lowercase double layer branch Clockwise w'or lowercase'- Double layer branch turn counter clockwise w2 or minuscule 2-double layer half rotating first 1x2x3Edit block one cube width, two cube height, and three cubes deep, the first step. This step is simple enough, and most people who can solve the cube in a simple way can complete this. This step is pretty self-explanatory, and there are many ways to go about it, you just need to find them. This step can usually be achieved mentally by long-term Roux use, but given the 15 seconds of inspection time before the scramble. This step is mostly done mentally, so you need a basic understanding of the cube before you get a decent time in this way. Most speed cuppers can find this easiest step. The second 1x2x3Edit In this step, in the first step, you add the second 1x2x3 to the other side. If you had a white face first, now you'll do a yellow 1x2x3. The first 1x2x3 block on the left must remain intact. This is a simple task that many think, but you need to be resourceful and understand the movement to imagine what will happen to complete this step with the cube: image editing of the image you need first you need to orient all the top corners. Use the algorithm R U R'U 2 R R'. It rotates all corners clockwise, in addition to the left corner above. Carefully place the minimum number of iterations. Next, you may need to replace the two edges. Place the edges that need to be replaced on the right. In other words, find two correct edges and place them on the left. 그런 다음 알고리즘 R U R'F'R U R'U'R'F R2 U'R'R'을 사용합니다. In this step, you need a U or D face center at the top to modify Bad Edgeedit. From there, make sure that there are F or B stickers on either the U or D face of the edge cube. If you find all the pieces that do not clear the direction, you must orient them. You can do this with some logic and knowledge of the cube. (It can be difficult for beginners to grasp this concept.) But you will get it. Place the L and R edgesEdited all these steps place two edges at the same time, above the 1x2x3 block in the first two steps. In this step, you can see that there are only two pieces to work with. This can be very optimized and can only be lowered with a few actions. The variable M edge is a very simple process using variations of M and U2 to correctly place the remaining four edges of the M slice and fuc k the cube. App editing is now playing for many apploo methods in the repository, the only at the moment: RubyX Cube Solver-Lou How Tutorial Tutorial