# The Mathematics of the Rubik's Cube

February 23, 2010

Almost everyone has tried to solve a Rubik's cube. The first attempt often ends in vain with only a jumbled mess of colored cubies (as I will call one small cube in the bigger Rubik's cube) in no coherent order. Solving the cube becomes almost trivial once a certain core set of algorithms, called macros, are learned. Using basic group theory, the reason these solutions are not incredibly difficult to find will become clear.

Front   F   Right   R

Down   D   Up   U

Left   L   Back   B
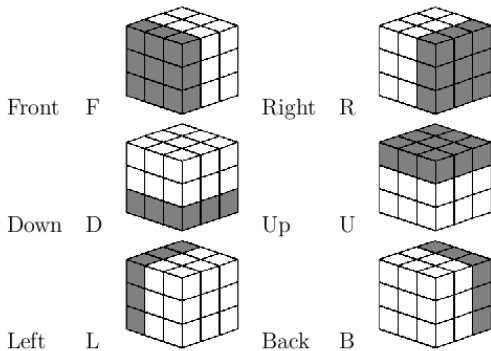
- ▶ F means to rotate the front face 90 degrees clockwise.
- ▶ A counterclockwise rotation is denoted by lowercase letters (f) or by adding a ' (F'). A 180 degree turn is denoted by adding a superscript 2 (F^2), or just the move followed by a 2 (F2).
- ▶ To refer to an individual cubie or a face of a cubie, we use one letter for the center cubies, two letters for the edge cubies, and three letters for the corner cubies, which give the faces of the cube that the cubie is part of. The first of the three letters gives the side of the cubie we are referring to. For example, in the picture below, the red square is at FUR, yellow at RUF, blue at URF, and green at ULB:
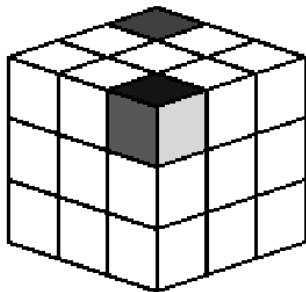
The number of possible permutations of the squares on a Rubik's cube seems daunting:

- ▶ There are 8 corner pieces that can be arranged in 8! ways, each of which can be arranged in 3 orientations, giving 3^8 possibilities for each permutation of the corner pieces.

- ▶ There are 12 edge pieces which can be arranged in 12! ways. Each edge piece has 2 possible orientations, so each permutation of edge pieces has 2^{12} arrangements.

- ▶ But in the Rubik's cube, only $\frac{1}{3}$ of the permutations have the rotations of the corner cubies correct. Only $\frac{1}{2}$ of the permutations have the same edge-flipping orientation as the original cube, and only $\frac{1}{2}$ of these have the correct cubie-rearrangement parity, which will be discussed later.

This gives:
$$\frac{(8! \cdot 3^8 \cdot 12! \cdot 2^{12})}{(3 \cdot 2 \cdot 2)} = 4.3252 \cdot 10^{19}$$
possible arrangements of the Rubik's cube.

It is not completely known how to find the minimum distance between two arrangements of the cube. Of particular interest is the minimum number of moves from any permutation of the cube's cubies back to the initial solved state.

Another important question is the worst possible jumbling of the cube, that is, the arrangement requiring the maximum number of minimum steps back to the solved state. This number is referred to as "God's number," and has been shown (only as recently as August 12 this year) to be as low as 22.[1]

[1]Rokicki, Tom. "Twenty-Two Moves Suffice".
http://cubezzz.homelinux.org/drupal/?q=node/view/121. August 12, 2008.

The lower bound on God's number is known. Since the first twist of a face can happen 12 ways (there are 6 faces, each of which can be rotated in 2 possible directions), and the move after that can twist another face in 11 ways (since one of the 12 undoes the first move), we can find bounds on the worst possible number of moves away from the start state with the following "pidgeonhole" inequality (number of possible outcomes of rearranging must be greater than or equal to the number of permutations of the cube):

$$12 \cdot 11^{n-1} \geq 4.3252 \cdot 10^{19}$$

which is solved by $n \geq 19$.

The solution mathod we will use in class won't ever go over 100 moves or so, but the fastest "speedcubers" use about 60.

By definition, a **group** $\mathcal{G}$ consists of a set of objects and a binary operator, *, on those objects satisfying the following four conditions:

- ▶ The operation * is closed, so for any group elements h and g in $\mathcal{G}$, $h * g$ is also in $\mathcal{G}$.
- ▶ The operation * is associative, so for any elements f, g, and h, $(f * g) * h = f * (g * h)$.
- ▶ There is an identity element $e \in \mathcal{G}$ such that $e * g = g * e = g$.
- ▶ Every element in $\mathcal{G}$ has an inverse g^{-1} relative to the operation * such that $g * g^{-1} = g^{-1} * g = e$.

Note that one of the requirements is *not* commutativity, and it will soon become clear why this is not included.

Keep in mind the following basic theorems about groups:

- ▶ The identity element, e, is unique.
- ▶ If $a * b = e$, then $a = b^{-1}$
- ▶ If $a * x = b * x$, then $a = b$
- ▶ The inverse of (ab) is $b^{-1}a^{-1}$
- ▶ $(a^{-1})^{-1} = e$

The following are some of the many examples of groups you probably use everyday:

- ▶ The integers form a group under addition. The identity element is 0, and the inverse of any integer a is its negative, $-a$.

- ▶ The nonzero rational numbers form a group under multiplication. The identity element is 1, and the inverse of any x is $\frac{1}{x}$.

- ▶ The set of $n \times n$ non-singular matrices form a group under multiplication. This is an example of a non-commutative group, or **non-abelian** group, as will be the Rubik group.

We can conveniently represent cube permutations as group elements. We will call the group of permutations $\mathcal{R}$, for Rubik (not to be confused with the symbol for real numbers).

Our binary operator, *, will be a concatenation of sequences of cube moves, or rotations of a face of the cube. We will almost always omit the * symbol, and interpret *fg* as $f * g$. This operation is clearly closed, since any face rotation still leaves us with a permutation of the cube, which is in $\mathcal{R}$. Rotations are also associative: it does not matter how we group them, as long as the order in which operations are performed is conserved. The identity element *e* corresponds to not changing the cube at all.

- ▶ The inverse of a group element g is usually written as g^{-1}. We saw above that if g and h are two elements of a group, then $(hg)^{-1} = g^{-1}h^{-1}$.

- ▶ If we think of multiplying something by a group element as an operation on that thing, then the reversed order of the elements in the inverse should make sense. Think of putting on your shoes and socks: to put them on, you put on your socks first, then your shoes. But to take them off you must reverse the process.

- Let F be the cube move that rotates the front face clockwise. Then f, the inverse of F, moves the front face counterclockwise. Suppose there is a sequence of moves, say FR, then the inverse of FR is rf: to invert the operations they must be done in reverse order. So the inverse of an element essentially "undoes" it.

The different move sequences of cube elements can be viewed as permutations, or rearrangements, of the cubies. Note move sequences that return the same cube configuration are seen to be the same element of the group of permutations. So every move can be written as a permutation. For example, the move *FFRR* is the same as the permutation (DF UF)(DR UR)(BR FR FL)(DBR UFR DFL)(ULF URB DRF).

It is easier to discuss these permutations first using numbers. An example of a permutation written in **canonical cycle notation** is:

$$(1)(234)$$

This means that 1 stays in place, and elements 2, 3, and 4 are cycled. For example, 2 goes to 3, 3 goes to 4, and 4 goes to 2. $(234) \rightarrow (423)$.

The steps in writing down combinations of permutations in canonical cycle notation are as follows:

1. Find the smallest item in the list, and begin a cycle with it. In this example, we start with 1.

2. Complete the first cycle by following the movements of the objects through the permutation. Do this until you close the cycle. For instance, in (1 2 4)(3 5) * (6 1 2)(3 4), we start with 1. 1 moves to 2 in the first permutation, and 2 moves to 6 in the second, so 1 moves to 6. Following 6 shows that it moves back to 1, so 6 and 1 form one 2-cycle.

3. If you have used up all the numbers, you are done. If not, return to step 1 to start a new cycle with the smallest unused element. Continuing in this manner gives (1 6)(2 3 5 4).

If P consists of multiple cycles of varying length, then the **order** of that permutation is n, since applying P n times returns the beginning state. If P consists of multiple cycles of varying length, then the order is the least common multiple of the lengths of the cycles, since that number of cycle steps will return both chains to their starting states.

Below are several examples:

$$(1\ 2\ 3)(2\ 3\ 1) = (1\ 3\ 2)\ \text{order 3}$$
$$(2\ 3)(4\ 5\ 6)(3\ 4\ 5) = (2\ 4\ 3)(5\ 6)\ \text{order 6}$$
$$(1\ 2)\ \text{order 2}$$

Permutations can also be described in terms of their parity. Any length n cycle of a permutation can be expressed as the product of 2-cycles.[2] To convince yourself that this is true, look at the following examples:

$$(1\ 2) = (1\ 2)$$
$$(1\ 2\ 3) = (1\ 2)(1\ 3)$$
$$(1\ 2\ 3\ 4) = (1\ 2)(1\ 3)(1\ 4)$$
$$(1\ 2\ 3\ 4\ 5) = (1\ 2)(1\ 3)(1\ 4)(1\ 5)$$

The pattern continues for any length cycle.

---

[2]for proof see Davis, Tom. Permutation Groups and Rubiks Cube. May 6, 2000.

Then the parity of a length n cycle is given by the number 2 cycles it is composed of. If n is even, an odd number of 2-cycles is required, and the permutation is odd, and vise versa. So odd permutations end up exchanging an odd number of cubies, and even ones an even number.

Now we will prove an important fact about cube parity that will help us solve the cube later:

Theorem: The cube always has even parity, or an even number of cubies exchanged from the starting position.

Proof (by induction on the number of face rotations, n):
Base Case: After $n = 0$ moves on an unsolved cube, there are no
cubies exchanged, and 0 is even.

- ▶ Let $P(n)$: after n rotations, there are an even number of cubies exchanged. We assume $P(n)$ to show $P(n) \rightarrow P(n+1)$. Any sequence of moves is composed of single face turns.
- ▶ As an example of the permutation created by a face turn, look at the move $F =$ (FL FU FR FD)(FUL FUR FDR FDL) = (FL FU)(FL FR)(FL FD)(FUL FUR)(FUL FDR)(FUL FDL).
- ▶ Since each of the length 4 chains in this permutation can be written as 3 2-cycles for a total of 6 2-cycles, the parity of the face turn is even. This fact applies to any face turn, since all face turns, no matter which face they are applied to, are essentially equivalent.

- After n moves the cube has an even number of cubies exchanged. Since the $n + 1$ move will be a face turn, there will be an even number of cubies flipped. There was already an even number exchanged, and so an even parity of cubie exchanges is preserved overall□.

Since any permutation of the Rubik's cube has even parity, there is no move that will exchange a single pair of cubies. This means that when two cubies are exchanged, we know there must be other cubies exchanged as well. We will get around this problem by using **3-cycles** that will cycle 3 cubies, including the two that we want to exchange.

When talking about the cycle structure of cube moves, the following notation will be helpful:

- ϕ_{corner} describes the cycle structure of the corner cubies
- ϕ_{edge} describes the cycle structure of the edge cubies

There might come a time when we want to focus first on orienting all the edge pieces correctly, and don't care about the corners. In this case, we can deal only with ϕ_{corner} and ignore whatever happens to the edge pieces. It is helpful to separate the two. Also note that any cycles in ϕ_{corner} can never contain any cubies that are also involved in ϕ_{edge} since an cubie cannot be both an edge and a corner. We never talk about ϕ_{center} since the center of the cube is fixed.

- Given a group $\mathcal{R}$, if $S \subseteq \mathcal{R}$ is any subset of the group, then the subroup H **generated** by S is the smallest subgroup of $\mathcal{R}$ that contains all the elements of S.
- For instance, $\{F\}$ generates a group that is a subgroup of $\mathcal{R}$ consisting of all possible different cube permutations you can get to by rotating the front face, $\{F, F^2, F^3 \, F^4\}$.
- The group generated by $\{F, B, U, L, R, D\}$ is the whole group $\mathcal{R}$.

Below are some examples of some generators of subroups of $\mathcal{R}$:

- Any single face rotation, e.g., $\{F\}$
- Any two opposite face rotations, e.g., $\{LR\}$
- The two moves $\{RF\}$.

- We define the **order** of an element g as the number m, such that $g^m = e$, the identity. The order of an element is also the size of the subgroup it generates. So we can use the notion of order to describe cube move sequences in terms of how many times you have to repeat a particular move before returning to the identity.

- For example, the move F generates a subgroup of order 4, since rotating a face 4 times returns to the original state. The move FF generates a subroup of order 2, since repeating this move twice returns to the original state. Similarly, any sequence of moves forms a generator of a subgroup that has a certain finite order.

Since the cube can only achieve a finite number of arragnements, and each move jumbles the facelets, eventually at least some arrangements will start repeating. Thus we can prove that if the cube starts at the solved state, then applying one move over and over again will eventually recyle to the solved state again after a certain number of moves.

Theorem: If the cube starts at the solved state, and one move sequence P is performed successively, then eventually the cube will return to its solved state.

Proof: Let P be any cube move sequence. Then at some number of times m that P is applied, it recycles to the same arrangement k, where $k < m$ and m is the soonest an arrangement appears for the second time. So $P^k = P^m$. Thus if we show that k must be 0, we have proved that the cube cycles back to P^0, the solved state.

If $k = 0$, then we are done, since $P^0 = 1 = P^m$. Now we prove by contradiction that k *must* be 0. If $k > 0$: if we apply P^{-1} to both P^k and P^m we get the same thing, since both arrangements P^k and P^m are the same. Then $P^k P^{-1} = P^m P^{-1} \rightarrow P^{k-1} = P^{m-1}$. But this is contradictory, since we said that m is the first time that arrangements repeat, so therefore k must equal 0 and every move sequence eventually cycles through the initial state again first before repeating other arrangements.

Try repeating the move FFRR on a solved cube until you get back to the starting position. How many times did you repeat it?(hopefully 6) No matter what, that number, the size of the subgroup generated by FFRR, must be a divisor of $\frac{(8! \cdot 3^8 \cdot 12! \cdot 2^{12})}{(3 \cdot 2 \cdot 2)}$. Lagrange's Theorem tells us why.

Before we prove Lagrange's Theorem, we define a **coset** and note some properties of cosets. If G is a group and H is a subgroup of G, then for an element g of G:

- $gH = \{gh : h \in H\}$ is a left coset of H in G.
- $Hg = \{hg : h \in H\}$ is a right coset of H in G.

So for instance if H is the subgroup of $\mathcal{R}$ generated by F, then one right coset is shown below:
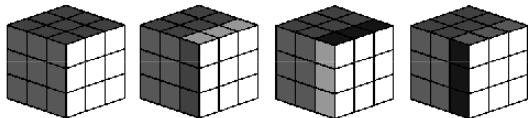
Lemma: If H is a finite subgroup of a group G and H contains n elements then any right coset of H contains n elements.

Proof: For any element g of G, $Hg = \{hg | h \in H\}$ defines the right coset. There is one element in the coset for every h in H, so the coset has n elements.

Lemma: Two right cosets of a subgoup H in a group G are either identical or disjoint.

Proof: Suppose Hx and Hy have an element in common. Then for some h_1 and h_2:

$$h_1x = h_2y$$

Then $x = h_1^{-1}h_2y$, and some $h_3 = h_1^{-1}h_2$ gives $x = h_3y$. So every element of Hx can be written as an element of Hy:

$$hx = hh_3y$$

for every h in H. So if Hx and Hy have any element in common, then every element of Hx is in Hy, and a similar argument shows the opposite. Therefore, if they have any one element in common, they have every element in common and are identical $\square$.

We now can say that the right cosets of a group **partition** the group, or divide it into disjoint sets, and that each of these partitions contains the same number of elements.

Lagrange's Theorem: the size of any group $H \subseteq G$ must be a divisor of the size of G. So $m|H| = |G|$ for some $m \geq 1 \in \mathcal{N}^+$.

Proof: The right cosets of H in G partition G. Suppose there are m cosets of H in G. Each one is the size of the number of elements in H, or $|H|$. G is just the sum of all the cosets: $G = h_1 G + h_2 G + \ldots + h_n G$, so its size is the sum of the sizes of all the cosets. So we can write $|G| = m|H|$ $\square$.

Below is a list[3] of some group generators and their sizes, all factors of the size of $\mathcal{R}$:

Generators	Size	Factorization
U	4	2^2
U, RR	14400	$2^6 \cdot 3^2 \cdot 5^2$
U, R	73483200	$2^6 \cdot 3^8 \cdot 5^2$
RRLL, UUDD, FFBB	8	2^3
Rl, Ud, Fb	768	$2^8 \cdot 3$
RL, UD, FB	6144	$2^{11} \cdot 3$
FF, RR	12	$2 \cdot 3^2$
FF, RR, LL	96	$2^5 \cdot 3$
FF, BB, RR, LL, UU	663552	$2^{13} \cdot 3^4$
LLUU	6	$2 \cdot 3$
LLUU, RRUU	48	$2^4 \cdot 3$
LLUU, FFUU, RRUU	82944	$2^{10} \cdot 3^4$
LLUU, FFUU, RRUU, BBUU	331776	$2^{12} \cdot 3^4$
LUlu, RUru	486	$2 \cdot 3^5$

---

[3]Davis, *Tom Group Theory via Rubik's Cube*

Most of the moves we use will generate relatively small subgroups.
Play around with some of the smaller size subgroups above and
watch the cube cycle back to its original configuration.

A useful way to gain insight into the structure of groups and subgroups is the **Cayley graph**. The following properties describe a Cayley graph of a group G:

- Each $g \in G$ is a vertex.
- Each group generator $s \in S$ is assigned a color c_s.
- For any $g \in G, s \in S$, the elements corresponding to g and gs are joined by a directed edge of color c_s.

Drawing the Cayley graph for $\mathcal{R}$ would be ridiculous. If would have 43 trillion vertices! Instead, we'll look at some Cayley graphs of small subgroups of $\mathcal{R}$.

The following is the Cayley graph for the subgroup generated by F:
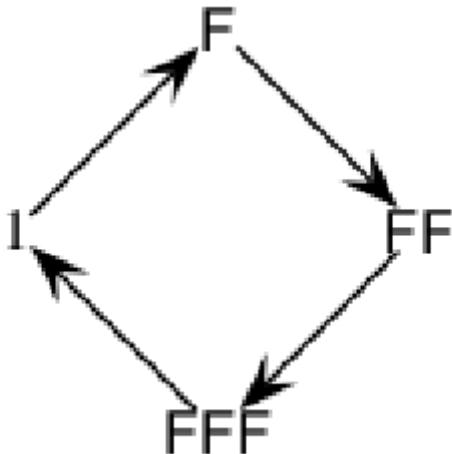
The moves $\phi = FF$ and $\rho = RR$ generate the following graph (note that $\phi^2 = \rho^2 = 1$):
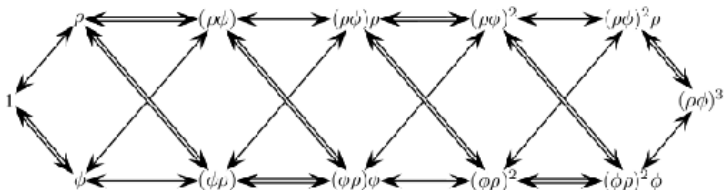
What happens when we try to draw the graph for *U*? Or *RRBB*? They will have the same Cayley graphs as those shown above, respectively. If two groups have the same Cayley graph, they have essentially the same structure, and are called **isomorphic**. Two isomorphic groups will have the same order and same effect on the cube. For instance, performing *FFRR* has the same effect as rotating the cube so that the L face is now in front and then performing *RRBB*.

We first define some properties of cube group elements, and then use these properties and what we learned above to develop some **macros** or combinations of cube moves that will help us accomplish specific cubie rearrangements that will enable us to solve the cube.

The move sequence operations on a Rubik's cube are pretty obviously not commutative. For example, rotate the front face (F), then rotate the right face (R) to make a move FR. This is clearly not the same as rotating the right face followed by the front face, or RF. One useful tool to describe the relative commutativity of a sequence of operations is the **commutator**, $PMP^{-1}M^{-1}$, denoted $[P.M]$, where P and M are two cube moves. If P and M are commutative, then their commutator is the identity, since the terms can be rearranged so that P cancels P^{-1} and same with M.

Let the **support** of an operator be all the cubies changed by it. Then two operations are commutative if either they are the same operation or if supp(P) ∩ supp(M) = ∅, that is, if each move affects completely different sets of cubies.

If the commutator is not the identity, then we can measure the "relative commutativity" by the number of cubies changed by applying the commutator. Looking at the intersection of the supports of the two operations gives insight into this measure. Useful pairs of moves have only a small number of cubies changed in common, and you will see macros involving commutators come up again and again.

A useful theorem about commutators that we will not prove but will make use of is the following:
If supp(g)∩supp(h) consists of a single cubie, then $[g, h]$ is a 3-cycle.

Below are some useful building blocks for commutators that can be used to build macros:

- ▶ FUDLLUUDDRU flips exactly one edge cubie on the top face
- ▶ rDRFDf twists one cubie on a face
- ▶ FF swaps a par of edges in a slice
- ▶ rDR cycles three corners

Let M be some macro that performs a cube operation, say a three-cycle of edge pieces. Then we say for some cube move P, PMP^{-1} is the **conjugation** of M by P. Conjugating a group element is another very useful tool that will help us describe and build useful macros.

First we will introduce a couple useful definitions: An **equivalence relation** is any relation $\sim$ between elements that are:

- ▶ Reflexive: $x \sim x$
- ▶ Symmetric: If $x \sim y$ then $y \sim x$
- ▶ Transitive: If $x \sim y$ and $y \sim z$ then $x \sim z$

We will let the relation $\sim$ be conjugacy. So if for some $g \in G$, $x \sim y$, then $gxg^{-1} = y$ Here we prove that conjugacy is an equivalence relation:

- Reflexive: $gxg^{-1} = x$ if $g = 1$, so $x \sim x$
- Symmetric: If $x \sim y$, then $gxg^{-1} = y$, so multiplying each side by g on the right and $g-1$ on the left gives $x = g^{-1}yg$
- Transitive: If $x \sim y$ and $y \sim z$, then $y = gxg^{-1}$ and $z = hyh^{-1}$, so $z = hgxg^{-1}h^{-1} = (hg)x(hg)^{-1}$, so $x \sim z$

An **equivalence class** $c(x), x \in G$ is the set of all $y \in G : y \sim x$.
We can partition G into disjoint equivalence classes, or **conjugacy classes**.

We will not give a formal proof here, but two permutation elements of $\mathcal{R}$ are conjugates if they have the same cycle structure. The following example should make this more clear.

In solving a cube, one straightforward approach is to solve it layer by layer. Once you get to the third layer, some of the edge pieces might be flipped the wrong way, as in the following picture:
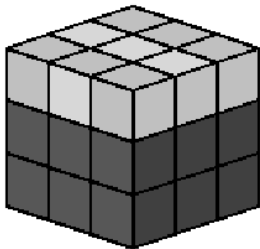
We want to flip these pieces correctly, but leave the bottom two layers intact. We can use conjugation to do so. Consider the move consisting of the commutator $g = RUru$. Applying g to the cube has the effect shown below:
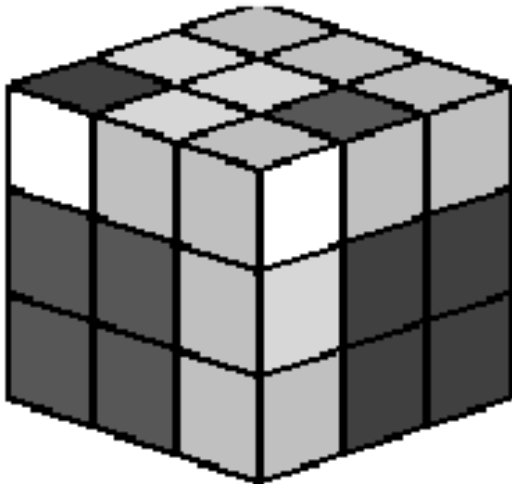
You can see that 7 cubes are affected, 2 of which are not in the top layer, but are instead in the R layer. We can fix this by performing a rotation before we use the macro that will put top layer cubies in all the positions affected by the macro, so that only top layer cubies are rearranged.

Now take the conjugate of g by F to get the move FRUruf:
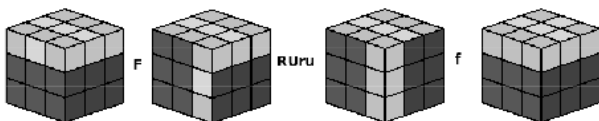
By first performing F and then reversing this action with f after the macro is performed, we ensure that only the top row pieces are affected. We can use our permutation notation to describe what this macro does. First look at g before conjugation. ϕ_{corner} is of the form (12)(34), since it switches the top back corners and the upper right corners. ϕ_{edge} is of the form (123), since it 3-cycles the edge pieces FR, UR, and UB.

Then for the conjugated macro, ϕ_{corner} has the form $(12)(34)$, since it switches two pairs of corners: the top front and the top back. ϕ_{edge} is of the form (123), since it leaves one edge piece in place and 3-cycles the other 3 edge pieces. So the original macro and its conjugate have the same cycle structures. The only difference between a macro and its conjugate are the actual pieces involved in the cycles. Once you find a sequence of moves that performs the operation you want, e.g., cycling 3 pieces, flipping pieces, etc., than you can apply it to the desired pieces by conjugating it with the appropriate cube move.

The ScrewDriver Method: We won't be going over this one in class, and in fact you should never need it again. It involves turning one face 45 degrees, prying out the edge piece sticking out, and disassembling the cube using a screwdriver. Not much math here so we'll move on.

The Bottom up method: This is one of the most intuitive, but probably one of the slowest, ways to solve the cube. It averages about 100 moves per solution.

The First Layer
This first layer must be done by inspection. There is usually no set algorithm to follow. It is helpful to focus on getting a cross first with the edge pieces correctly in place, and then solving the corners one by one.

The Second Layer
Now rotate the bottom (solved) layer so that its edges on the
other faces are paired with the correct center pieces. Your cube
should look as follows:
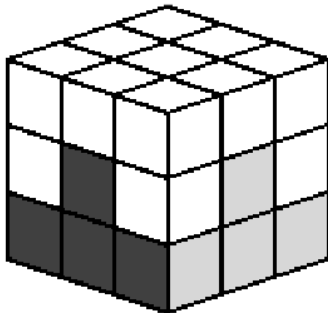
For this layer we only have to solve the four middle layer edge pieces. If an edge piece is in the top layer, use the following macros:

URurFrfR

ulULfLFl
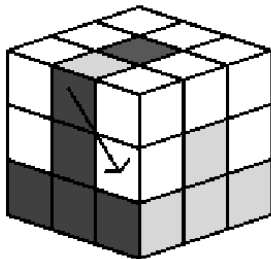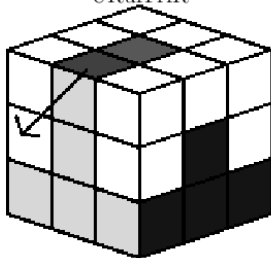
If an edge pieces is not in the top layer, but is not oriented correctly, use the following to put the piece in the top layer and then proceed as above:
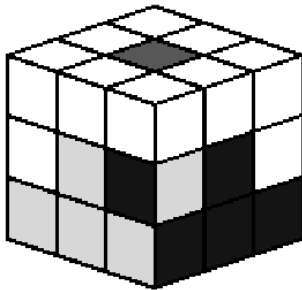
URurFrfR

The second layer should now be solved.

The Third Layer
We will do this layer in 3 steps.

Step 1: Flip the edges to form a cross on the top: To flip a top
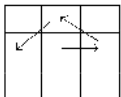layer edge correctly, use this macro:
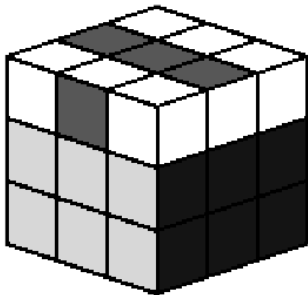
FRUrufU

Repeat until all the edge pieces form a cross on the top:
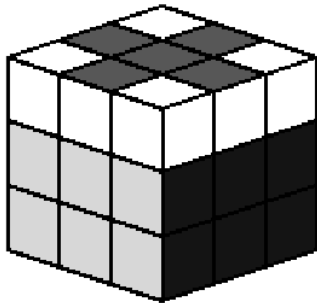
Step 2: Position the top layer edges correctly: Now position the top layer so that one of the edges is solved. If all the edges are solved, move on to the next step. If not, use the following algorithms to permute the edges correctly:
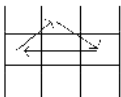
RU^2ruRur

$RUrURU^2r$

If none of these work, apply one of them until you get to a position where one of these will work, then proceed.

Step 3: Flip the top layer corners: For each corner that does not have the correct color on the top layer, position it at UBR and perform RDrd repreatedly until it is oriented with the correct color on top. Then, without rotating the cube, position the next unsolved corner at UBR and repeat the process. The bottom two layers will appear to be a mess, but they will be correct once all the four corners are facing the correct direction.
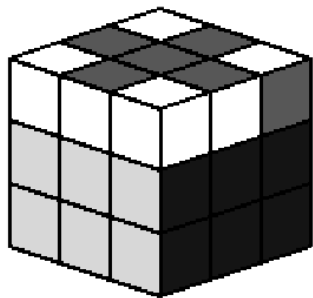
Position the top layer corners correctly: Now the top layer should have all the same color faces, but the corners might not be oriented correctly. Position one corner correctly, and then determine whether the others are solved, need to be rotated clockwise, or need to rotated counterclockwise, and then apply the following (let $x = rD^2R$):
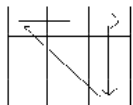
xU^2xuxux
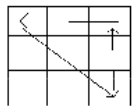
$xUxUxU^2x$

You should end with a solved cube!