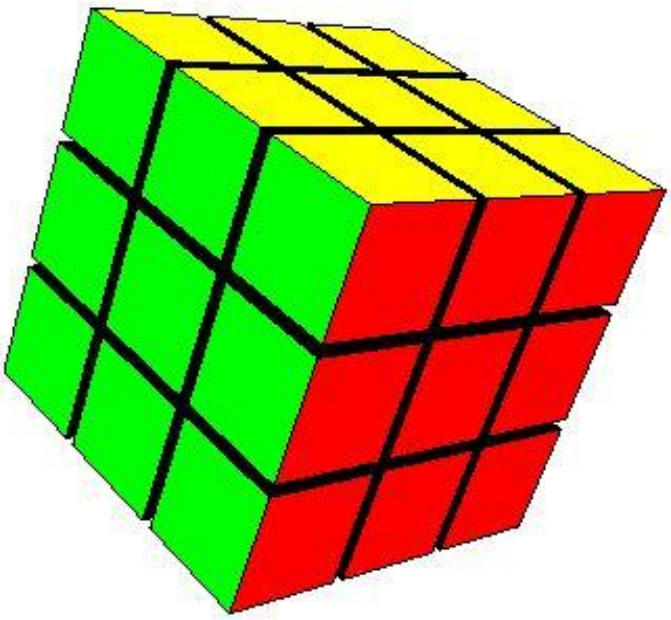
Rubik's Cube



Contents

1 Introduction

Rubik's cube is a famous toy in the puzzle category. It has initiated a lot of research into the mathematical structures of the cube.

There are of course software packages that can help in analysing the cube.
Two of them are:

- Cube Explorer v 5.00, download at http://kociemba.org/cube.htm
  So you don't need an actual Rubik's cube.
- GAP, http://www.gap-system.org/ , Groups, Algorithms, Programming
  Can be of great help.

1.1 Some History

Ernő Rubik (Hungarian pronunciation: [ˈrubik ˈɛrno], born July 13, 1944) is a Hungarian inventor, architect and professor of architecture.
He is best known for the invention of mechanical puzzles including Rubik's Cube (1974).
He is known to be an introvert, barely accessible and hard to contact or to get hold of for autographs.
He typically does not attend speedcubing events.

The puzzle was licensed by Rubik to be sold by Ideal Toy Corp. in 1980 and won the German Game of the Year special award for Best Puzzle that year. As of January 2009, 350 million cubes have sold worldwide making it the world's top-selling puzzle game. It is widely considered to be the world's best-selling toy.

Speedcubing is a competition in solving the Rubik Cube as fast as possible and requires a high level of dexterity coupled with some higher brain functions. Typical results are 5-10 seconds and considering that the diameter of the Cayley graph has been determined to 20 the impressive factor is high.
Cayley graph is explained in 5.17 below.

At www.speedcubing.com one can find speedcubing events, world records, etc
A small excerpt from the list of 2011 events:

2011 Dec 25 Tianjin Open 2011 China, Tianjin Tianjin University of Technology and Education
2011Dec 24Hokuriku Eve 2011Japan, Kanazawa Kanazawa Kinrousha Plaza
2011Dec 17MIT Fall 2011USA, Cambridge, Massachusetts Massachusetts Institute of Technology
2011Dec 17-18Puy de Dome Open 2011France, DalletSalle Polyvalente
2011Dec 17León Winter 2011Mexico, León, Guanajuato Colegio Nuevo Continente
2011Dec 17Mantua Winter 2011Italy, Mantova Politecnico di Milano - Polo territoriale di Mantova
2011Dec 17Dutch Nationals 2011Netherlands, Zaandam De Koekfabriek

The original cube consists of 3x3x3 (often named 3x3 cube) small cubes, but that is only the starting point. On the net one can e.g. find 100x100x100 Rubik cubes, probably implemented in software, doing it mechanically, non-virtual, seems a daunting task. Up to 5x5x5 has been manufactured in hardware.

2 Introduction

Seeing the Rubik's cube for the first time most people probably ask How-the-&#@-can-you-possibly-solve-this-puzzle ?
The next some people do, is take the Rubik cube apart to see how it works internally ?
And then some (smaller number of) people start to work out the mathematical properties of the cube ?
The following is some information about the first and the second of the above questions.

3 Notation

The normal 3x3x3 Rubik's cube is composed of 27 small cubes, which are typically called "cubies."
26 of these cubies are visible (if you take your cube apart, you'll find that the 27th cubie doesn't actually exist).

When working with the Rubik's cube, it's helpful to have a systematic way of referring to the individual cubies. Although it seems natural to use the colors of a cubie, it is actually more useful to have names which describe the locations of the cubies.

3.1 Name of cubies

The cubies in the corners are called, appropriately enough, "corner" cubies.
Each corner cubie has 3 visible faces, and there are 8 corner cubies.
The cubies with two visible faces are called "edge" cubies.
There are 12 edge cubies.
Finally, the cubies with a single visible face are called "center cubies," and there are 6 center cubies.

3.2 Name of faces

Now, let's name the 6 faces of the Rubik's cube.
Following the notation developed by David Singmaster, they are called
right (r), left (l), up (u), down (d), front (f), and back (b).
The advantage of this naming scheme is that each face can be referred to by a single letter.

3.3 Cubie names using face references

Corner cubie
To name a corner cubie, we simply list its visible faces in clockwise order.
For instance, the cubie in the upper, right, front corner is written urf.
Oriented and unoriented cubies
Of course, it could also called rfu or fur. When orientation of the cubie is important, the distinction is important. In that case the cubie is an "oriented cubie".
That is, the oriented cubies urf, rfu, and fur are different. In situations when the order of faces is not important, the cubie is an "unoriented cubie." That is, the unoriented cubies urf, rfu, and fur are the same.

Similarly, to name edge and center cubies, we will just list the visible faces of the cubies.

Edge cubie
For instance, the edge cubies in the front face are fr, fl, fu, fd

Center cubie
For instance, the cubie in the center of the front face is just called f, because its only visible face lies on the front of the cube.

Cubicles
These are labeled the same way as cubies, but describe the space in which the cubie lives. Thus, if the Rubik's cube is in the start configuration (that is, the Rubik's cube is solved), then each cubie lives in the cubicle of the same name (the urf cubie lives in the urf cubicle, the f cubie lives in the f cubicle, and so on). If a face is rotated, the cubicles don't move, but the cubies do. Notice, however, that when a face is rotated, all center cubies stay in their cubicles.

3.4 Name of moves

Finally, we want to give names to some moves of the Rubik's cube.
The most basic move one can do is to rotate a single face.
We will let R denote a clockwise rotation of the right face (looking at the right face, turn it 90 degrees clockwise). R' or R- is the same move but counter clockwise.
Similarly, we will use the capital letters L, U, D, F, and B to denote clockwise twists of the corresponding faces. More generally, we will call any sequence of these 6 face twists a "move" of the Rubik's cube. For instance, rotating the right face counterclockwise is a move which is the same as doing R three times.
One example: FD'R2 means first front face 1/4 turn clockwise, then down face 1/4 turn counter-clockwise, then right face 1/4 turn clockwise two times.

A couple of things are immediately clear.
- First, we already observed that the 6 basic moves keep the center cubies in their cubicles. Since any move is a sequence of these 6 basic moves, that means that every move of the Rubik's cube keeps the center cubies in their cubicles.
- Any move of the Rubik's cube puts corner cubies in corner cubicles and edge cubies in edge cubicles. It is impossible for a corner cubie to ever live in an edge cubicle or for an edge cubie to live in a corner cubicle.

4 Number of states of the Cube

There are 8! ways to arrange the 8 corner cubies. Seven can be oriented independently, and the orientation of the eighth depends on the preceding seven, giving 3^7 possibilities.
There are 12!/2 ways to arrange the 12 edges, since an odd permutation of the corners implies an odd permutation of the edges as well. Eleven edges can be flipped independently, with the flip of the twelfth depending on the preceding ones, giving 2^{11} possibilities.
All in all
$2^{12} \cdot 3^8 \cdot 12! \cdot 8! / 12 = 43252003274489856000 = 2^{27} \cdot 3^{14} \cdot 5^3 \cdot 7^2 \cdot 11 = \sim 4 \cdot 10^{19}$

5 Groups and Rubik's cube

Let's call the Rubik's Cube group RC-G.

Making the Rubik's Cube into a Group

We can make the set of moves of the Rubik's cube into a group, which we will denote (RC-G, *). The elements of RC-G will be all possible moves of the Rubik's cube (for example, one possible move is a clockwise turn of the top face followed by a counter clockwise turn of the right face). Two moves will be considered the same if they result in

the same configuration of the cube (for instance, twisting a face clockwise by 180 degrees is the same as twisting the same face counter clockwise by 180 degrees).

The group operation will be defined like this:
if M1 and M2 are two moves, then M1* M2 is the move where you first do M1 and then do M2.
We need to show the properties of a group.

• RC-G is certainly closed under * since, if M1 and M2 are moves, M1*M2 is a move as well.

• If we let e be the "empty" move (that is, a move which does not change the configuration of the Rubik's cube at all), then M * e means "first do M, then do nothing."
This is certainly the same as just doing M, so M * e = M = e* M. So, (RC-G, *) has an identity element.

• If M is a move, we can reverse the steps of the move to get a move M0. Then, the move M * M0 means "first do M, then reverse all the steps of M." This is the same as doing nothing, so M * M0 = e, so M0 is the inverse of M. Therefore, every element of RC-G has a right inverse and with the same arguments can be shown to have a left inverse.
If * is associative, shown in next bullet, then the left and right inverse are the same element. Suppose we have group elements g, h a left inverse and k a right inverse, then h = h * e = h * (g * k) = (h * g) * k = e * k = k

• Finally, we must show that * is associative. A move can be defined by the change in configuration it causes. In particular, a move is determined by the position and orientation it puts each cubie in.
If C is an oriented cubie, we will write M(C) for the oriented cubicle that C ends up in after we apply the move M, with the faces of M(C) written in the same order as the faces of C. That is, the first face of C should end up in the first face of M(C), and so on. For example, the move D puts the ur cubie in the br cubicle, with the u face of the cubie lying in the b face of the cubicle and the r face of the cubie lying in the r face of the cubicle.
Thus, we write D(ur) = br.
First, let's investigate what a sequence of two moves does to the cubie. If M1 and M2 are two moves, then M1 * M2 is the move where we first do M1 and then do M2. The move M1 moves C to the cubicle M1(C). The move M2 then moves it to M2(M1(C)). Therefore, (M1 * M2)(C) = M2(M1(C)).
To show that * is associative, we need to show that (M1 * M2) * M3 = M1 * (M2 * M3) for any moves M1, M2, and M3.
This is the same as showing that (M1 * M2) * M3 and M1 * (M2 * M3) do the same thing to every cubie.
That is, we want to show that [(M1 * M2) * M3](C) = [M1 * (M2 * M3)](C) for any cubie C.
We know from our above calculation that
[(M1 * M2) * M3](C) = M3([M1 * M2](C)) = M3(M2(M1(C))).
On the other hand, [M1 * (M2 * M3)](C) = (M2 * M3)(M1(C)) = M3(M2(M1(C))).So, (M1 * M2) * M3 = M1 * (M2 * M3).
Thus, * is associative.

All put together, (RC-G, *) is a group.

RC-G is a subgroup of the symmetric groupS_{48} , (S48),
which has order |S48| = 48! =
$$124139155925360726708622890473733750385214863546777600000000000$$ =
= 2^46*3^22*5^10*7^6*11^4*13^3*17^2*19^2*23^2*29*31*37*41*43*47 =~ 1,2*10^61

The order, sometimes called period, of an element of a group is the smallest positive integer m such that a^m = e

Theorem. (Cauchy) Let p be a prime dividing |G|. There is a g ϵ G of order p.

Theorem. (Lagrange) Let n be an integer not dividing |G|. There does not exist a g ϵ G of order n.

David Singmaster states that the maximal order in the Rubik's cube group is 1260 = 2^2*3^2*5*7

Definition: The center of a group G is the subgroup Z(G) of all elements which commute with every element of G: Z(G) = {z ϵ G | z * g = g * z, for all g ϵ G}

The **center** of G consists of exactly two elements, the identity and the "superflip" move which has the effect of flipping over every "edge", leaving all the corners alone and leaving all the sub cubes in their original position.
One move for the superflip is = RLFBUDRLFBUF2MF2U'M2B2M'B2UM2D
more information can be found in chapter 5-4 of [W.D.J]

Obviously, concepts like cyclic, permutation, symmetry, subgroups etc play a big part in the structure of Rubik's cube and its group.

The Rubik's cube group is not abelian, e.g. RB ≠ BR, though subgroups may be.

5.1 Equivalence relation

A given binary relation ~ on a set A is said to be an equivalence relation iff it is reflexive, symmetric and transitive. Equivalently, for all a, b and c in A:
a ~ a. (Reflexivity)
if a ~ b then b ~ a. (Symmetry)
if a ~ b and b ~ c then a ~ c. (Transitivity)

Equivalence relations on sets partitions a set (into disjoint sets).

5.2 Conjugacy class

Definition Suppose G is a group. Two elements a and b of G are called conjugate if there exists an element g in G with

$$gag^{-1} = b$$

It can be readily shown that conjugacy is an equivalence relation and therefore partitions G into equivalence classes. This means that every element of the group belongs to precisely one conjugacy class, and the classes Cl(a) and Cl(b) are equal if and only if a and b are conjugate, and disjoint otherwise.
The equivalence class that contains the element a in G is

$$Cl(a) = \{gag^{-1} : g \in G\}$$

and is called the conjugacy class of a.

The class number of G is the number of distinct (nonequivalent) conjugacy classes.

Example

The symmetric group S3, consisting of all 6 permutations of three elements, has three conjugacy classes:
- no change (abc → abc)
- interchanging two (abc → acb, abc → bac, abc → cba)
- a cyclic permutation of all three (abc → bca, abc → cab)

5.3 Permutation group

Definition A permutation group is a group G whose elements are permutations of a given set M, and whose group operation is the composition of permutations in G

(which are thought of as bijective functions from the set M to itself).

The relationship is often written as (G,M).
Note that the group of all permutations of a set is the symmetric group.
The term permutation group is usually restricted to mean a subgroup of the symmetric group.

Permutations are often written i cycle notation.

5.4 Cycle notation and row notation

Let us consider S6 and let a permutation θ be (using **two-row notation**)

θ : $\begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 \\ 1 & 2 & 5 & 6 & 4 & 3 \end{pmatrix}$

Only giving the second row is called **one-line/one-row notation**

The effect of θ is 3->5, 5->4, 4->6 and 6->3 and leaving 1,2 unaltered.
θ is called a cycle and is in **cycle notation** denoted (3, 5, 4, 6)

More generally

Definition Let S be a finite set, and
$a_1, \ldots, a_k, \quad k \geq 2$
be distinct elements of S.
The expression $(a_1 \ \ldots \ a_k)$ denotes the cycle σ whose action is
$a_1 \mapsto a_2 \mapsto a_3 \mapsto \ldots \mapsto a_k \mapsto a_1.$
For each index i, σ(a_i) = a_i + 1, where a_k + 1 is taken to mean a_1.

There are k different expressions for the same cycle. The following all represent the same cycle:

$$(a_1\ a_2\ a_3\ \ldots\ a_k) = (a_2\ a_3\ \ldots\ a_k\ a_1) = \cdots = (a_k\ a_1\ a_2\ \ldots\ a_{k-1}).$$

A 1-element cycle such as (3) is the identity permutation. The identity permutation can also be written as an empty cycle, "()".

Every cycle is a permutation. It is clear that not every permutation is a cycle. Nor is the product of two cycles necessarily a cycle (Note, evaluated from left to right)
e.g. in S4 :

(1,2)(3,4) = $\begin{pmatrix} 1 & 2 & 3 & 4 \\ 2 & 1 & 4 & 3 \end{pmatrix}$, which is not a cycle.

Can we express every element of Sn as a product of cycles ?
Yes, if the cycles are **disjoint cycles.**

Theorem Every element of Sn can be written as the product of **disjoint cycles**
(cycles $(a_1, \ldots, a_m)$ and $(b_1, \ldots, b_k)$ are disjoint if the a_i and b_j are distinct, i.e. $\{a_1, \ldots, a_m\} \cap \{b_1, \ldots, b_k\} = \phi$ "=empty set")

For proof, see e.g. [BBBC theorem 5.26]

Corollary If π ϵ Sn and $a_1, \ldots, a_m$ are chosen as in the proof of the above theorem, i.e. $a_2 = a_1\pi$, ... and $a_m\pi = a_1$ and $a_1, \ldots, a_m$ distinct, then
π = $(a_1, \ldots, a_m)$ τ
where $a_\tau = a_\pi$ if $a \notin \{a_1, \ldots, a_m\}$, while a_j τ = a_j for j = 1, ..., m

This corollary provides a method of computing the decomposition of an elements π ϵ Sn into the product of disjoint cycles.

Example:

π = $\begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 11 & 12 \\ 3 & 4 & 2 & 1 & 8 & 7 & 9 & 11 & 12 & 10 & 5 & 6 \end{pmatrix}$

since $1_\pi \neq 1$, we may take 1 for a_1. Then $a_1 = 1$, $a_2 = 3$, $a_3 = 2$, $a_4 = 4$, $a_5 = 1$.
So m = 4, and by the corollary

$\pi = (1, 3, 2, 4) \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 11 & 12 \\ 1 & 2 & 3 & 4 & 8 & 7 & 9 & 11 & 12 & 10 & 5 & 6 \end{pmatrix}$

and so on, to finally get

$\pi = (1, 3, 2, 4)(5, 8, 11)(6, 7, 9, 12)$

5.5 Alternating group

Definition An alternating group is the group of even permutations of a finite set.
The alternating group on the set {1,...,n} is called the alternating group of degree n, or the alternating group on n letters and denoted by An or Alt(n).

Basic properties

For n > 1, the group An is the commutator subgroup of the symmetric group Sn with index 2 and has therefore n!/2 elements.
It is the kernel of the signature group homomorphism sgn : Sn → {1, −1} explained under symmetric group.
The group An is abelian if and only if n ≤ 3 and simple if and only if n = 3 or n ≥ 5. A5 is the smallest non-abelian simple group, having order 60, and the smallest non-solvable group.

Conjugacy classes

As in the symmetric group, the conjugacy classes in An consist of elements with the same cycle shape. However, if the cycle shape consists only of cycles of odd length with no two cycles the same length, where cycles of length one are included in the cycle type, then there are exactly two conjugacy classes for this cycle shape.
Examples:
- the two permutations (123) and (132) are not conjugates in A3, although they have the same cycle shape, and are therefore conjugate in S3
- the permutation (123)(45678) is not conjugate to its inverse (132)(48765) in A8, although the two permutations have the same cycle shape, so they are conjugate in S8.

Example Alternating group A4 of order 4!/2 = 12, has the following multiplication table, using cycle notation :

	(1)	(123)	(124)	(134)	(234)	(132)	(142)	(143)	(243)	(12)(34)	(13)(24)	(14)(23)
(1)	(1)	(123)	(124)	(134)	(234)	(132)	(142)	(143)	(243)	(12)(34)	(13)(24)	(14)(23)
(123)	(123)	(132)	(13)(24)	(234)	(12)(34)	(1)	(143)	(14)(23)	(124)	(134)	(243)	(142)
(124)	(124)	(14)(23)	(142)	(13)(24)	(123)	(134)	(1)	(243)	(12)(34)	(143)	(132)	(234)
(134)	(134)	(124)	(12)(34)	(143)	(13)(24)	(14)(23)	(234)	(1)	(132)	(123)	(142)	(243)
(234)	(234)	(13)(24)	(134)	(14)(23)	(243)	(142)	(12)(34)	(123)	(1)	(132)	(143)	(124)
(132)	(132)	(1)	(243)	(12)(34)	(134)	(123)	(14)(23)	(142)	(13)(24)	(234)	(124)	(143)
(142)	(142)	(234)	(1)	(132)	(14)(23)	(13)(24)	(124)	(12)(34)	(143)	(243)	(134)	(123)
(143)	(143)	(12)(34)	(123)	(1)	(142)	(243)	(13)(24)	(134)	(14)(23)	(124)	(234)	(132)
(243)	(243)	(143)	(14)(23)	(124)	(1)	(12)(34)	(132)	(13)(24)	(234)	(142)	(123)	(134)
(12)(34)	(12)(34)	(243)	(234)	(142)	(124)	(143)	(134)	(132)	(123)	(1)	(14)(23)	(13)(24)
(13)(24)	(13)(24)	(142)	(143)	(243)	(132)	(234)	(123)	(124)	(134)	(14)(23)	(1)	(12)(34)
(14)(23)	(14)(23)	(134)	(132)	(123)	(143)	(124)	(243)	(234)	(142)	(13)(24)	(12)(34)	(1)

5.6 Symmetric group

Not to be confused with Symmetry group, see e.g. wikipedia.org

The symmetric group Sn on a finite set of n symbols is the group whose elements are **all the permutations** of the n symbols, and whose **group operation** is the **composition of such permutations**, which are treated as bijective functions from the set of symbols to itself. Since there are n! possible permutations of a set of n symbols,
it follows that |Sn| = n!. (symmetric groups **can** be defined on infinite sets as well, they behave quite differently).

Definition The symmetric group on a finite set X is the group whose elements are all bijective functions from X to X and whose group operation is that of function composition. For finite sets, "permutations" and "bijective functions" refer to the same operation, namely rearrangement.
The symmetric group of degree n is the symmetric group on the set X = { 1, 2, ..., n }.

The symmetric group on a set X is denoted in various ways including SX, $\mathfrak{S}_X$, ΣX, and Sym(X).

If X is the set { 1, 2, ..., n }, then the symmetric group on X is also denoted Sn, $\mathfrak{S}_n$, Σn, and Sym(n).

The group operation in a symmetric group is function composition, denoted by the symbol ∘or simply by juxtaposition of the permutations. The composition $f \circ g$ of permutations f and g, pronounced "f after g", maps any element x of X to f(g(x)).

Concretely, let

$$f = (1\ 3)(4\ 5) = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 \\ 3 & 2 & 1 & 5 & 4 \end{pmatrix}$$ and

$$g = (1\ 2\ 5)(3\ 4) = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 \\ 2 & 5 & 4 & 3 & 1 \end{pmatrix}.$$

The "matrix" with the two rows defines a mapping X->X, the ()() is cycle notation which is described in a separate chapter.

Applying f after g maps 1 first to 2 and then 2 to itself; 2 to 5 and then to 4; 3 to 4 and then to 5, and so on. So composing f and g gives

$$fg = f \circ g = (1\ 2\ 4)(3\ 5) = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 \\ 2 & 4 & 5 & 1 & 3 \end{pmatrix}.$$

A cycle of length L = k·m, taken to the k-th power, will decompose into k cycles of length m: For example (k = 2, m = 3),

$$(1\ 2\ 3\ 4\ 5\ 6)^2 = (1\ 3\ 5)(2\ 4\ 6).$$

Verification of group axioms

To check that the symmetric group on a set X is indeed a group, it is necessary to verify the group axioms of associativity, identity, and inverses.
- The operation of function composition is always associative.
- The trivial bijection that assigns each element of X to itself serves as an identity for the group.
- Every bijection has an inverse function that undoes its action, and thus each element of a symmetric group does have an inverse.

5.7 Symmetries and Symmetry group

Not to be confused with Symmetric group.

The symmetry group of an object (image, signal, etc.) is the group of all isometries under which it is invariant with composition as to the operation.

5.7.1 Symmetry groups and Rubik's Cube

There are 48 geometric rotations, reflections etc., which map a cube onto a cube. Subsets of these maps define symmetry subgroups of the cube.
There are 33 essential different types of subgroups:

Oh, O, Td, D3d, Th, C3v, T, D4h, D3, D4, D2d(face), C4v, C4h, D2h(edge), D2d(edge), S6, D2h(face), C2v(a1), C2v(b), C2h(b), D2(edge), C4, D2(face), S4, C2h(a), C2v(a2), C3, Cs(b), C2(b), C2(a), Cs(a), Ci, C1(identity)

5.8 Cyclic groups

A cyclic group is a group that can be generated by a single element, in the sense that the group has an element g (called a "generator" of the group) such that, when written multiplicatively, every element of the group is a power of g (a multiple of g when the notation is additive).
Definition A group *G* is called cyclic if there exists an element *g* in *G* such that
$G = \langle g \rangle = \{ g^n \mid n \text{ is an integer} \}$.

Since any group generated by an element in a group is a subgroup of that group, showing that the only subgroup of a group G that contains *g* is *G* itself suffices to show that G is cyclic.
The fundamental **theorem** of cyclic groups states that if G is a cyclic group of order n then every subgroup of G is cyclic.

Theorem There exist cyclic groups of all orders, finite and infinite. Any two cyclic groups of the same order are isomorphic. (One therefore often talk about ***the*** cyclic group of order m, or ***the*** infinite cyclic group, or sometimes ***the*** infinite cycle) [BBBC chapter 4]

Cyclic groups are abelian.

Example $\mathbf{Z_m} = \{ e, g^1, g^2, ..., g^{m-1} \}$, m = 2, 3, ... and $g^m = e$

Example Z_m^3 is a 3-tuple of elements of $\mathbf{Z_m}$, elements belong to $\mathbf{Z_m} \times \mathbf{Z_m} \times \mathbf{Z_m}$
Z_m^3 is an abelian group, but is **not** a cyclic group.

5.8.1 Moves of 3x3x3 Rubik's cube

Using notation due to Singmaster [Si] allows us to check that the puzzle is in fact a permutation puzzle. The Rubik's cube has 6 sides, or "faces", each of which has 3 x 3 = 9 "facets", for a total of 54 facets. We label these facets 1, 2, ..., 54 as follows:

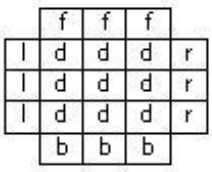
```
                +--------------+
                |  1    2    3 |
                |               |
                |  4    u    5 |
                |               |
                |  6    7    8 |
+---------------+--------------+--------------+--------------+
|  9   10   11 | 17   18   19 | 25   26   27 | 33   34   35 |
|               |              |              |              |
| 12    l   13 | 20    f   21 | 28    r   29 | 36    b   37 |
|               |              |              |              |
| 14   15   16 | 22   23   24 | 30   31   32 | 38   39   40 |
+---------------+--------------+--------------+--------------+
                | 41   42   43 |
                |              |
                | 44    d   45 |
                |              |
                | 46   47   48 |
                +--------------+
```

Then the generators, corresponding to the six faces of the cube, may be written in disjoint cycle notation as:
F = (17 19 24 22)(18 21 23 20)(6 25 43 16)(7 28 42 13)(8 30 41 11)
B = (33 35 40 38)(34 37 39 36)(3 9 46 32)(2 12 47 29)(1 14 48 27)
L = (9 11 16 14)(10 13 15 12)(1 17 41 40)(4 20 44 37)(6 22 46 35)
R = (25 27 32 30)(26 29 31 28)(3 38 43 19)(5 36 45 21)(8 33 48 24)
U = (1 3 8 6)(2 5 7 4)(9 33 25 17)(10 34 26 18)(11 35 27 19)
D = (41 43 48 46)(42 45 47 44)(14 22 30 38)(15 23 31 39)(16 24 32 40)

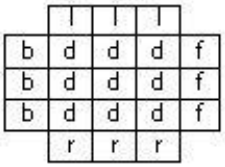
All moves are permutations, i.e. the Rubik's cube is a permutation puzzle.

Using another notation

Down face d, front face f, left face l, right face r, back face b



After the move D:



Move D in cycle notation: D = (dlf dfr drb dbl)(df dr db dl).

5.9 Subgroup definition

Definition: A nonempty subset H of a group (G, *) is called a subgroup of G if (H, *) is a group.
Notation: If G is a group then we will denote the statement "H is a subgroup of G" by **H < G** or ***H ≤ G***

Often useful: Let (G, *) be a group. A nonempty subset H of G is a subgroup of (G, *) iff, for every a, b ϵ H, $a * b^{-1}$ ϵ H.

5.10 Cosets

Definition Let X and Y be subsets of a group G, the set XY is defined as:
XY = {g | g = xy, x ϵ G and y ϵ G}

- Let A, B, C be subsets of a group G, then A(BC) = (AB)C
- If H is a subset of G and f,g ϵ G, then (fg)H = f(gH), H(fg) = (Hf)g and (fH)g = f(Hg)

Definition Let H be a subgroup of a group G.
Then a **right coset** of H in G is a subset of the form
Hg = {x | x = hg, h ϵ H } for some g ϵ G
And a **left coset** of H in G is a subset of the form
gH = {x | x = gh, h ϵ H } for some g ϵ G

Theorem Let H be a subgroup of a group G. Then the right (left) cosets of H in G form a partition of H in G, i.e. the union of all the right (left) cosets of H in G is G itself and any pair of distinct cosets has empty intersection. [BBBC]

Suppose aH and bH are two cosets, then they are disjoint or identical.

5.11 Direct and semi-direct product

Definition Given groups G and H, the **direct product** G × H is defined as follows:
The elements of G × H are ordered pairs (g, h), where g ∈ G and h ∈ H. That is, the set of elements of G × H is the Cartesian product of the sets G and H.
The binary operation on G × H is defined componentwise:

(g1, h1) · (g2, h2) = (g1 · g2, h1 · h2)

The resulting algebraic object satisfies the axioms for a group. Specifically:

Associativity	The binary operation on G × H is indeed associative.
Identity	The direct product has an identity element, namely (1G, 1H), where 1G is the identity element of G and 1H is the identity element of H.
Inverses	The inverse of an element (g, h) of G × H is the pair (g−1, h−1), where g−1 is the inverse of g in G, and h−1 is the inverse of h in H.

Definition Given any two groups N and H (not necessarily subgroups of a given group) and a group homomorphism φ : H → Aut(N),
there is a new group $N \rtimes_\varphi H$ (or simply $N \times_\varphi H$), called the semidirect product of N and H with respect to φ, defined as follows.

As a set, $N \rtimes_\varphi H$ is the cartesian product N × H.
Multiplication of elements in $N \rtimes_\varphi H$ is determined by the homomorphism φ.
The operation is
$*\colon (N \rtimes_\varphi H) \times (N \rtimes_\varphi H) \to N \rtimes_\varphi H$
defined by $(n_1, h_1) * (n_2, h_2) = (n_1 \varphi_{h_1}(n_2), h_1 h_2)$ for n1, n2 in N and h1, h2 in H.
This defines a group in which the identity element is (eN, eH) and the inverse of the element (n, h) is (φh–1(n–1), h–1). Pairs (n, eH) form a normal subgroup isomorphic to N, while pairs (eN, h) form a subgroup isomorphic to H.
The full group is a semidirect product of those two subgroups in the sense given above.

Definition Let G be a group with identity element e, N a normal subgroup of G (i.e., N ◁ G) and H a subgroup of G.
The following statements are equivalent and If one (and therefore all) of these

statements hold, we say that G is a semidirect product of N and H, written $G = N \rtimes H,$

- G = NH and N ∩ H = {e}.
- G = HN and N ∩ H = {e}.
- Every element of G can be written as a unique product of an element of N and an element of H.
- Every element of G can be written as a unique product of an element of H and an element of N.
- The natural embedding H → G, composed with the natural projection G → G / N, yields an isomorphism between H and the quotient group G / N.
- There exists a homomorphism G → H which is the identity on H and whose kernel is N.

One also says G splits over N, G is a semidirect product of H acting on N
In order to avoid ambiguities, it is advisable to specify which of the two subgroups is normal.

5.12 Normal subgroup

A subgroup, N, of a group, G, is called a normal subgroup if it is invariant under conjugation; that is, for each element n in N and each g in G, the element gng−1 is still in N. We write

$$N \triangleleft G \;\Leftrightarrow\; \forall n \in N, \forall g \in G \;, gng^{-1} \in N.$$

For any subgroup, the following conditions are equivalent to normality. Therefore any one of them may be taken as the definition:

For all g in G, gNg−1 ⊆ N.

For all g in G, gNg−1 = N.

The sets of left and right cosets of N in G coincide.

For all g in G, gN = Ng.

N is a union of conjugacy classes of G.

There is some homomorphism on G for which N is the kernel.

The last condition accounts for some of the importance of normal subgroups. They are a way to internally classify all homomorphisms defined on a group. For example, a non-identity finite group is simple if and only if it is isomorphic to all of its non-identity homomorphic images, a finite group is perfect if and only if it has no normal subgroups of prime index, and a group is imperfect if and only if the derived subgroup is not supplemented by any proper normal subgroup.

Examples

- The subgroup {e} consisting of just the identity element of G and G itself are always normal subgroups of G. The former is called the trivial subgroup, and if these are the only normal subgroups, then G is said to be simple.
- The center of a group is a normal subgroup.
- The commutator subgroup is a normal subgroup.

- More generally, any characteristic subgroup (see 5.13) is normal, since conjugation is always an automorphism.
- All subgroups N of an abelian group G are normal, because gN = Ng. A group that is not abelian but for which every subgroup is normal is called a Hamiltonian group.
- The translation group in any dimension is a normal subgroup of the Euclidean group; for example in 3D rotating, translating, and rotating back results in only translation; also reflecting, translating, and reflecting again results in only translation (a translation seen in a mirror looks like a translation, with a reflected translation vector). The translations by a given distance in any direction form a conjugacy class; the translation group is the union of those for all distances.
- In the Rubik's Cube group, the subgroup consisting of operations which only affect the corner pieces is normal, because no conjugate transformation can make such an operation affect an edge piece instead of a corner. By contrast, the subgroup consisting of turns of the top face only is not normal, because a conjugate transformation can move parts of the top face to the bottom and hence not all conjugates of elements of this subgroup are contained in the subgroup.

5.13 Characteristic subgroup

A characteristic subgroup is a subgroup that is invariant under all automorphisms of the parent group. Because conjugation is an automorphism, every characteristic subgroup is normal, though not every normal subgroup is characteristic. Examples of characteristic subgroups include the commutator subgroup and the center of a group

Definition

A characteristic subgroup of a group G is a subgroup H that is invariant under each automorphism of G. That is,

φ(H) = H for every automorphism φ of G (where φ(H) denotes the image of H under φ).

The statement “H is a characteristic subgroup of G” is written H char G

Characteristic vs. normal

If G is a group, and g is a fixed element of G, then the conjugation map

$$x \mapsto gxg^{-1}$$

is an automorphism of G (known as an inner automorphism). A subgroup of G that is invariant under all inner automorphisms is called normal. Since a characteristic subgroup is invariant under all automorphisms, every characteristic subgroup is normal.

Not every normal subgroup is characteristic.
One examples:
Let H be a group, and let G be the direct product H × H. Then the subgroups {1} × H and H × {1} are both normal, but neither is characteristic. In particular, neither of these subgroups is invariant under the automorphism (x, y) → (y, x) that switches the two factors.

5.14 Subgroups of RC-G

What are the subgroups of the Rubik's cube group? It turns out that there are too many to list but some of the subgroups can be given as examples.

Theorem (Lagrange): Let H be a subgroup of a finite group G. Then |H| divides |G|.

It is remarkable that several "familiar" groups may be embedding into the Rubik's cube group, and hence be regarded as a subgroup of the cube group. For example, it is shown in chapter 8.3 of [W.D.J] how to embed the group of quaternions Q = {1, -1, i, -i, j, -j, k, -k} inside the Rubik's cube group.

5.14.1 The structure of the Rubik's cube group

We consider two subgroups of RC-G: First the group of cube orientations, Co, which leaves every block fixed, but can change its orientation. This group is a normal subgroup of RC-G. It can be represented as the normal closure of some operations that flip a few edges or twist a few corners. For example, it is the normal closure of the following two operations:

BR'D2RB'U2BR'D2RB'U2 , (twist two corners)

RUDB2U2B'UBUB2D'R'U' , (flip two edges).

For the second group we take RC-G permutations, Cp, which can move the blocks around, but leaves the orientation fixed. For this subgroup there are more choices, depending on the precise way you fix the orientation. One choice is the following group, given by generators (the last generator is a 3 cycle on the edges):

C_p = [U2, D2, F, B, L2, R2, R2U'FB'R2F'BU'R2]

Since Co is a normal subgroup, the intersection of Co and Cp is the identity, and their product is the whole cube group, it follows that the cube group RC-G is the semi-direct product of these two groups. That is

$$G = C_o \rtimes C_p.$$

Next we can take a closer look at these two groups. Co is an abelian group, it is

$$\mathbb{Z}_3^7 \times \mathbb{Z}_2^{11}.$$

The group of cube permutations, Cp, is a little more complicated. It has the following two normal subgroups, the group of even permutations on the corners A8 and the group of even permutations on the edges A12. Complementary to these two groups we can take a permutation that swaps two corners and swaps two edges. We obtain that

$$C_p = (A_8 \times A_{12}) \rtimes \mathbb{Z}_2.$$

Putting all the pieces together we get that the cube group RC-G is isomorphic to

$$(\mathbb{Z}_3^7 \times \mathbb{Z}_2^{11}) \rtimes ((A_8 \times A_{12}) \rtimes \mathbb{Z}_2).$$

5.14.2 Examples of subgroups of the Rubik's cube group

5.14.2.1 Simple move subgroup

Let H be the subgroup of the Rubik's cube group generated by the basic move R: H =< R >. Then H $\cong$ C_4 (where C_4 denotes the cyclic group of order 4).

5.14.2.2 The squares subgroup

Let G denote the subgroup of the Rubik's cube group generated by the squares of the basic moves:

G := < U2, D2, R2, L2, F2, B2 >

called the squares group. The order of this group is 2^13*3^4
A more detailed treatment of this subgroup can be found in [W.D.J] chap. 12.1.

5.15 Valid states for the 3x3x3 Rubik's cube

5.15.1 Configuration space

Notation remark: Z/nZ , the set of integers mod n

The configuration of the Rubik's cube is determined by four pieces of data:
• the positions of the corner cubies
• the positions of the edge cubies
• the orientations of the corner cubies
• the orientations of the edge cubies

The **first** can be described by an element **r** of S_8 (i.e., the element of S_8 which moves the corner cubies from their start positions to the new positions).
The **second** can be described by an element **s** of S_{12}.

Now, we will see how to understand the third and fourth. The basic idea is to fix a "starting orientation" and a systematic way of writing down how a given orientation differs from this starting orientation. This is mostly just a matter of notation.
For details, see [J.C.] chapter 6.

Orientation of corner cubies, 8 of them
We will describe the orientations of the corner cubies like this: for any i between 1 and 8, find the cubicle face labeled i, let v_i be the number of the cubie face living in this cubicle face. We write **v** for the ordered 8-tuple (v_1, . . . , v_8). Notice that we can think of each v_i as counting the number of clockwise twists the cubie i is away from having its 0 face in the numbered face of the cubicle. But a cubie that is 3 twists away is oriented the same way as a cubie that is 0 twists away. Thus, we should think of the v_i as being elements of Z/3Z. So, **v** is an 8-tuple of elements of Z/3Z, we write **v** ∈ $(\mathrm{Z}/3\mathrm{Z})^8$

Orientation of edge cubies, 12 of them

Each edge cubie now has a face lying in a numbered cubicle face, label this cubie face 0, and label the other face of the cubie 1. Then, let w_i be the number of the cubie face in the cubicle face numbered i. This defines **w** ∈ $(\mathrm{Z}/2\mathrm{Z})^{12}$.

Thus, any configuration of the Rubik's cube can be written as an ordered 4-tuple (**r, s, v, w**), where **r** ∈ S_8 , **s** ∈ S_{12}, **v** ∈ $(\mathrm{Z}/3\mathrm{Z})^8$, and **w** ∈ $(\mathrm{Z}/2\mathrm{Z})^{12}$
The 4-tuple must obey the rules of "Second fundamental theorem of cube theory" below.

5.15.2 2nd fundamental theorem of cube theory

In reference [W.D.J] this theorem states some 'conservation' laws for the cube
Theorem (#202 in the book)

(Second fundamental theorem of cube theory) A 4-tuple (v, r, w, s) as above (r ∈ S_8, s ∈ S_{12}, v ∈ C_3^8 , w ∈ C_2^{12}) corresponds to a possible position of the Rubik's cube if and only if
(a) sgn(r) = sgn(s) ("equal parity as permutations")
(b) v_1 + ... + v_8 ≡ 0 (mod 3) ("conservation of total twists")
(c) w_1 + ... + w_{12} ≡ 0 (mod 2) ("conservation of total flips")

For details, look up in the reference. It is just interesting to mention these kinds of invariants.

5.16 Presentation of RC-G

One method of defining a group is by a presentation. One specifies a set S of generators so that every element of the group can be written as a product of powers of some of these generators, and a set R of relations among those generators.

We then say G has presentation $\langle S \mid R \rangle$.

Problem : Find

(a) a set of generators for RC-G of minimal cardinality,

(b) a set of relations for RC-G of minimal cardinality,

(c) an expression for each such generator as a word in the basic moves R, L, U, D, F, B.

The part (a) is known: there are 2 elements which generate RC-G [Si].
Part (b) is not known (though Dan Hoey's post of Dec 17, 1995 to the cube-lover's list may describe the best known results [CL]. He suggests that RC-G has a set X of 5 generators and a set Y of 44 relations such that the total length of all the reduced words in Y is 605).

5.17 Cayley graph

In mathematics, a **Cayley graph**, also known as a **Cayley colour graph**, **Cayley diagram**, **group diagram**, or **colour group** is a graph that encodes the abstract structure of a group. Its definition is suggested by **Cayley's theorem** (named after Arthur Cayley), and uses a specified, usually finite, set of generators for the group. It is a central tool in combinatorial and geometric group theory.

Definition

Suppose that *G* is a group and *S* is a generating set. The Cayley graph Γ = Γ(*G*,*S*) is a colored directed graph constructed as follows:

- Each element *g* of *G* is assigned a vertex: the vertex set *V*(Γ) of Γ is identified with *G*
- Each generator *s* of *S* is assigned a color c_s
- For any $g \in G, s \in S$, the vertices corresponding to the elements *g* and *gs* are joined by a directed edge of colour c_s. Thus the edge set *E*(Γ) consists of pairs of the form (*g*,*gs*), with $s \in S$ providing the color.

In geometric group theory, the set S is usually assumed to be finite, symmetric (i.e. S = S − 1) and not containing the identity element of the group. In this case, the uncolored Cayley graph is an ordinary graph: its edges are not oriented and it does not contain loops (single-element cycles) iff $1 \notin S$.

The full Cayley graph for the Rubik's Cube group has so far not been found. It's diameter has been determined to 20, playfully called God's Number, and some facts about the graph have been determined.

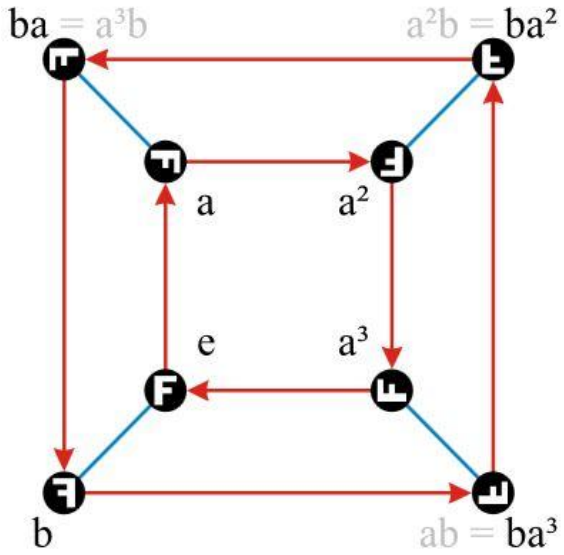
At the time of writing the number of positions for different distances is:

Distance	Count of Positions
0	1
1	18
2	243
3	3,240
4	43,239
5	574,908
6	7,618,438
7	100,803,036
8	1,332,343,288
9	17,596,479,795
10	232,248,063,316
11	3,063,288,809,012
12	40,374,425,656,248
13	531,653,418,284,628
14	6,989,320,578,825,358
15	91,365,146,187,124,313
16	about 1,100,000,000,000,000,000
17	about 12,000,000,000,000,000,000
18	about 29,000,000,000,000,000,000
19	about 1,500,000,000,000,000,000
20	about 300,000,000

Example of a Cayley graph using dihedral group D4

The Cayley graph of the group D_4 can be derived from the group presentation

$$\langle a, b | a^4 = b^2 = e, ab = ba^3 \rangle.$$



5.18 Cayley's Theorem

Cayley's Theorem states that every group *G* is isomorphic to a subgroup of the symmetric group acting on *G*. This can be understood as an example of the **group action** of *G* on the elements of *G*.

A permutation of a set *G* is any bijective function taking *G* onto *G* and the set of all such functions forms a group under function composition, called *the symmetric group on G*, and written as Sym(*G*).

Cayley's theorem puts all groups on the same footing, by considering any group (including infinite groups such as (***R***,+)) as a permutation group of some underlying set. Thus, theorems which are true for permutation groups are true for groups in general.

Proof of theorem

Where *g* is any element of *G*, consider the function $f_g : G \to G$, defined by $f_g(x) = g*x$. By the existence of inverses, this function has a two-sided inverse, $f_{g^{-1}}$. So multiplication by *g* acts as a bijective function. Thus, f_g is a permutation of *G*, and so is a member of Sym(*G*).

The set $K = \{f_g : g \in G\}$ is a subgroup of Sym(*G*) which is isomorphic to *G*. The fastest way to establish this is to consider the function $T : G \to \mathrm{Sym}(G)$ with $T(g) = f_g$ for every *g* in *G*. *T* is a group homomorphism because (using "•" for composition in Sym(*G*)):

$$(f_g \cdot f_h)(x) = f_g(f_h(x)) = f_g(h * x) = g * (h * x) = (g * h) * x = f_{(g*h)}(x),$$

for all *x* in *G*, and hence:

$$T(g) \cdot T(h) = f_g \cdot f_h = f_{(g*h)} = T(g * h).$$

The homomorphism *T* is also injective since $T(g) = \mathrm{id}_G$ (the identity element of Sym(*G*)) implies that *g*x = x* for all *x* in *G*, and taking *x* to be the identity element *e* of *G* yields *g = g*e = e*. Alternatively, *T* is also injective since, if *g*x=g'*x* implies *g=g'* (by post-multiplying with the inverse of *x*, which exists because *G* is a group).

Thus *G* is isomorphic to the image of *T*, which is the subgroup *K*.

T is sometimes called the *regular representation of G*.

5.19 Group action

Definition
If $(G, \cdot)$ is a group and X is a set, then a (left) group action of G on X is a binary operator:

$$\circ : G \times X \to X$$

that satisfies the following two axioms:

Associativity

$$(g \cdot h) \circ x = g \circ (h \circ x), \ \forall g, h \in G, \ x \in X;$$

Identity

$$e \circ x = x, \ \forall x \in X.$$

The set X is called a (left) G-set. The group G is said to act on X (on the left).
In a similar way a (right) group action can be defined.
Any right action has an equivalent left action, thus only left actions can be considered without any loss of generality.

5.20 Homomorphisms arising from group actions

Theorem/Lemma. Let G be a group and X a finite set. If G acts on X (on the left, resp. on the right) then there is a homomorphism G -> Sx given by g -> ϕ_g .
Conversely, if ϕ : G -> Sx is a homomorphism then ϕ(g) : X -> X defines a (left, resp. right) action of G on X.

Let RC-G be the Rubik's cube group generated by the basic moves R, L, U, D, F, B.
For each move g ϵ RC-G, let ϱ(g) be the corresponding permutation of the set of vertices V of the cube and let σ(g) be the corresponding permutation of the set of edges E of the cube. Let Sn denote the symmetric group on n letters and identify S-vertices with S8, S-edges with S12. Then

(a) ϱ: RC-G -> S8 is a homomorphism,

(b) σ: RC-G -> S12 is a homomorphism.

5.20.1 Some more advanced results

Definition. Let G act on a set X. We call the action k-tuply transitive if for each pair of ordered k-tuples (x1, x2, ..., xk), (y1, y2, ..., yk) of elements belonging to X there is a g ϵ G such that $y_i = \phi_g(x_i)$ for each 1 <= i <= k.

The following result is one illustration of how unique the symmetric group and alternating group are.

Theorem. If k > 5 and G is a group acting k-transitively on a finite set X then G is isomorphic to Sm or to An, for some m>= k or some n >= k +2.
Conversely, Sn acts n-transitively on {1, 2, ..., n} and
An acts (n - 2) - transitively on {1, 2, ..., n}.

Corollary. (a) The Rubik's cube group G acts 6-transitively on the corners, leaving the edges fixed. It acts 8-transitively on the corners but may permute two edges.
(b) The Rubik's cube group G acts 10-transitively on the edges, leaving the corners fixed. It acts 12-transitively on the edges but may permute two corners.

6 About solution of Rubik's cube

6.1 Length of solution for Rubik's Cube

There are two common ways to measure the length of a solution.
The first is to count the number of quarter turns.
The second is to count the number of face turns.
A move like F2 (a half turn of the front face) would be counted as 2 moves in the quarter turn metric and as only 1 turn in the face turn metric.

6.1.1 History of lower bound

It can be proven by counting arguments that there exist positions needing at least 18 moves to solve. To show this, first count the number of cube positions that exist in total, then count the number of positions achievable using at most 17 moves. It turns out that the latter number is smaller.

This argument was not improved upon for many years. Also, it is not a constructive proof , it does not exhibit a concrete position that needs this many moves.

It was conjectured that the so-called superflip would be a position that is very difficult. A Rubik's Cube is in the superflip pattern when each corner and edge piece is in the correct position, but each edge piece is incorrectly oriented.
In 1992, a solution for the superflip with 20 face turns was found by Dik T. Winter and its 'minimality' was shown in 1995 by Michael Reid , providing a new lower bound for the diameter of the cube group.
Also in 1995, a solution for superflip in 24 quarter turns was found by Michael Reid, with its 'minimality' proven by Jerry Bryan.

6.1.2 History of upper bound

Every solver of the Cube uses an algorithm, a sequence of steps for solving the cube. One algorithm might use a sequence of moves to solve the top face, then another sequence of moves to position the middle edges, and so on.
There are many different algorithms, varying in complexity and number of moves required, but those that can be memorized by a mortal typically require less than forty moves.

The first upper bounds were based on the 'human' algorithms.
By combining the worst-case scenarios for each part of these algorithms, the typical upper bound was found to be around 100.

The breakthrough was found by Morwen Thistlethwaite. Details of Thistlethwaite's Algorithm were published in March of 1981 Scientific American by Douglas Hofstadter.
The approaches to the cube that lead to algorithms with very few moves are based on group theory and on extensive computer searches.
Thistlethwaite's idea was to divide the problem into subproblems. Where algorithms up to that point divided the problem by looking at the parts of the cube that should remain fixed, he divided it by restricting the type of moves you could execute.
In particular,
First he divided the cube group into the following chain of subgroups:

G0 = <L,R,F,B,U,D>	This group contains all possible positions of the Rubik's cube.
G1 = <L,R,F,B,U2,D2>	This group contains all positions that can be reached

	(from the solved state) with quarter turns of the left, right, front and back sides of the Rubik's cube, but only double turns of the up and down sides.
G2 = <L,R,F2,B2,U2,D2>	In this group, the positions are restricted to ones that can be reached with only double turns of the front, back, up and down faces and quarter turns of the left and right faces.
G3 = <L2,R2,F2,B2,U2,D2>	Positions in this group can be solved using only double turns on all sides
G4 = {I}	Solved state of the cube.

(strategy: "The cube is now solved by moving from group to group, using only moves in the current group, for example, a scrambled cube likely lies in group G0. A look up table of possible permutations is used that uses quarter turns of all faces to get the cube into group G1. Once in group G1, quarter turns of the up and down faces are disallowed in the sequences of the look-up tables, and the tables are used to get to group G2, and so on, until the cube is solved")
Next he prepared tables for each of the right coset spaces G[i+1]\Gi.
For each element he found a sequence of moves that took it to the next smaller group.
After these preparations he worked as follows.
A random cube is in the general cube group G0. **Next** he found this element in the right coset space G1\G0. He applied the corresponding process to the cube. This took it to a cube in G1. **Next** he looked up a process that takes the cube to G2, next to G3 and finally to G4.

Although the whole cube group G0 is very large (~4.3×10^19), the right coset spaces G1\G0, G2\G1, G3\G2 and G3 are much smaller. The coset space G2\G1 is the largest and contains only 1082565 elements.
The number of moves required by this algorithm is the sum of the largest process in each step. In the original version this was 52.

Thistlethwaite's algorithm was improved by Herbert Kociemba in 1992. He reduced the number of intermediate groups to only two:
G0 = <L,R,F,B,U,D> , G1 = <L,R,F2,B2,U2,D2> , G2 = {I}.
As with Thistlethwaite's Algorithm, he would search through the right coset space G1\G0 to take the cube to group G1. Next he searched the optimal solution for group G1. The searches in G1\G0 and G1 were both done with a method equivalent to IDA*. The search in G1\G0 needs at most 12 moves and the search in G1 at most 18 moves, as Michael Reid showed in 1995. By generating also suboptimal solutions that take the cube to group G1 and looking for short solutions in G1, you usually get much shorter overall solutions. Using this algorithm solutions are typically found of fewer than 21 moves, though there is no proof that it will always do so.

In 1995 Michael Reid proved that using these two groups every position can be solved in at most 29 face turns, or in 42 quarter turns.

This result was improved by Silviu Radu in 2005 to 40.

Using these group solutions combined with computer searches will generally quickly give very short solutions. But these solutions do not always come with a guarantee of their minimality. To search specifically for minimal solutions a new approach was needed.

In 1997 Richard Korf announced an algorithm with which he had optimally solved random instances of the cube. Of the ten random cubes he did, none required more than 18 face turns. The method he used is called IDA* and is described in [REK]. Korf describes this method as follows:
IDA* is a depth-first search that looks for increasingly longer solutions in a series of iterations, using a lower-bound heuristic to prune branches once a lower bound on their length exceeds the current iterations bound.

It works roughly as follows. First he identified a number of subproblems that are small enough to be solved optimally.
He used: The cube restricted to only the corners, not looking at the edges
The cube restricted to only 6 edges, not looking at the corners nor at the other edges.
The cube restricted to the other 6 edges.
Clearly the number of moves required to solve any of these subproblems is a lower bound for the number of moves you will need to solve the entire cube.
Given a random cube C, it is solved as iterative deepening. First all cubes are generated that are the result of applying 1 move to them. That is C * F, C * U, … Next, from this list, all cubes are generated that are the result of applying two moves. Then three moves and so on. If at any point a cube is found that needs too many moves based on the upper bounds to still be optimal it can be eliminated from the list.
Although this algorithm will always find optimal solutions there is no worst case analysis. It is not known how many moves this algorithm might need. An implementation of this algorithm can be found in [REK].

In 2006, Silviu Radu further improved his methods to prove that every position can be solved in at most 27 face turns or 35 quarter turns.

Daniel Kunkle and Gene Cooperman in 2007 used a supercomputer to show that all unsolved cubes can be solved in no more than 26 moves (in face-turn metric). Instead of attempting to solve each of the billions of variations explicitly, the computer was programmed to bring the cube to one of 15,000 states, each of which could be solved within a few extra moves. All were proved solvable in 29 moves, with most solvable in 26. Those that could not initially be solved in 26 moves were then solved explicitly, and shown that they too could be solved in 26 moves.

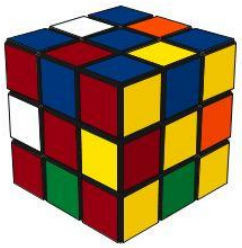
Tomas Rokicki reported in a 2008 computational proof that all unsolved cubes could be solved in 25 moves or fewer. This was later reduced to 23 moves. In August 2008 Rokicki announced that he had a proof for 22 moves. In 2009, Tomas Rokicki proved that 29 moves in quarter turn metric is enough to solve any scrambled cube.
Finally, in 2010, Tomas Rokicki, Herbert Kociemba, Morley Davidson, and John Dethridge gave the final computer-assisted proof that all cube positions could be solved with a maximum of 20 face turns.

6.1.3 Summary on bounds

There are many algorithms to solve scrambled Rubik's Cubes. The maximum number of face turns needed to solve any instance of the Rubik's cube is 20. This number is also known as the diameter of the Cayley graph of the Rubik's cube group and if often called God's number. An algorithm that solves a cube in the minimum number of moves is known as God's algorithm.

Superflip, the first position proven to require 20 moves, God's number, is

e.g. RLU2FU'DF2R2B2LU2F'B'UR2DF2UR2U , The result of the superflip is:



With about 35 CPU-years of idle computer time donated by Google, a team of researchers has essentially solved every position of the Rubik's Cube, and shown that no

position requires more than twenty moves. Any twist of any face is considered to be one move, i.e. 20 turns in the face metric.

It took fifteen years after the introduction of the Cube to find the first position that provably requires twenty moves to solve. It took an additional fifteen years to prove that twenty moves suffice for all positions.

Table with history of the search for God's number

Date	Lower bound	Upper bound	Gap	Notes and Links
July 1981	18	52	34	Morwen Thistlethwaite proves 52 moves suffice.
Dec 1990	18	42	24	Hans Kloosterman improves this to 42 moves.
May 1992	18	39	21	Michael Reid shows 39 moves is always sufficient.
May 1992	18	37	19	Dik Winter lowers this to 37 moves just one day later!
Jan 1995	18	29	11	Michael Reid cuts the upper bound to 29 moves by analyzing Kociemba's two-phase algorithm.
Jan 1995	20	29	9	Michael Reid proves that the "superflip" position (corners correct, edges placed but flipped) requires 20 moves.
Dec 2005	20	28	8	Silviu Radu shows that 28 moves is always enough.
Apr 2006	20	27	7	Silviu Radu improves his bound to 27 moves.
May 2007	20	26	6	Dan Kunkle and Gene Cooperman prove 26 moves suffice.
Mar 2008	20	23	3	Tomas Rokicki cuts the upper bound to 25 moves.
Apr 2008	20	23	3	Tomas Rokicki and John Welborn reduce it to only 23 moves.
Aug	20	22	2	Tomas Rokicki and John Welborn continue

2008				down to 22 moves.
Jul 2010	20	20	0	Tomas Rokicki, Herbert Kociemba, Morley Davidson, and John Dethridge prove that God's Number is exactly 20

6.1.4 How to find God's number

The group of people that found God's number:
Tomas Rokicki, a programmer from Palo Alto, California
Herbert Kociemba, a math teacher from Darmstadt, Germany
Morley Davidson, a mathematician from Kent State University
John Dethridge, an engineer at Google in Mountain View.

How did they solve all 43,252,003,274,489,856,000 positions of the Cube?

$$|G| = \frac{1}{12}\, 8!\, 3^8\, 12!\, 2^{12} = 43{,}252{,}003{,}274{,}489{,}856{,}000$$

Here is their own story:

We partitioned the positions into 2,217,093,120 sets of 19,508,428,800 positions each.
We reduced the count of sets we needed to solve to 55,882,296 using symmetry and set covering.
We did not find optimal solutions to each position, but instead only solutions of length 20 or less.
We wrote a program that solved a single set in about 20 seconds.
We used about **35 CPU years** to find solutions to all of the positions in each of the 55,882,296 sets.

Partitioning

We broke the problem down into 2,217,093,120 smaller problems, each comprising 19,508,428,800 different positions. Each of these subproblems was small enough to fit in the memory of a modern PC, and the way we broke it down (mathematically, using cosets of the group generated by {U,F2,R2,D,B2,L2}, or more concisely, cosets of H) allowed us to solve each set rapidly.

Symmetry

If you take a scrambled Cube and turn it upside down, you have not made it any more difficult. It will still take the same number of moves to solve. Instead of solving both of these positions, you can simply solve one, and then turn the solution upside down for the other. There are 24 different ways you can orient the Cube in space, and another factor of two using a mirror, for a total reduction of a factor of about 48 in the number of positions that need solving.
Using similar symmetry arguments and by finding a solution to a large "set cover" problem, we were able to reduce the number of sets that needed solving from 2,217,093,120 down to 55,882,296.

Good vs. Optimal Solutions

	Random positions	Cosets of H
Optimally	0.36	2,000,000
20 moves or less	3,900	1,000,000,000

Solution rate, in positions/second

An optimal solution to a position is one that requires no more moves than is required (remark quoting Einstein: "make it as simple as possible, but not simpler" :-)
Since a position that required 20 moves was already known, we did not need to optimally solve every position. We just needed to find a solution of 20 moves or less for each sequence. This is substantially easier. The table above shows the rate a good desktop PC has when solving random positions.

Fast Coset Solving Program

Using a combination of mathematical tricks and careful programming, we were able to solve a complete coset of H, either optimally, or with sequences of twenty moves or less, on a single desktop PC, at the rates shown in the above table.

Lots of Computers

Finally, we were able to distribute the 55,882,296 cosets of H among a large number of computers at Google and complete the computation in just a few weeks. Google does not release information on their computer systems, but it would take a good desktop PC (Intel Nehalem, four-core, 2.8GHz) 1.1 billion seconds, or about 35 CPU years, to perform this calculation.

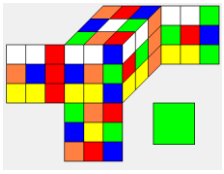
What are the Hardest Positions?

We have known for fifteen years that there are positions that require 20 moves. We have just proved that there are none that require more.

Distance-20 positions are both rare and plentiful. They are rarer than one in a billion positions, yet there are probably more than one hundred million such positions. We do not yet know exactly how many there are. The table below gives the count of positions at each distance. For distances 16 and greater, the number given is just an estimate. Our research has confirmed the prior results for entries 0 through 14 below, and the entry for 15 is a new result, which has since been independently confirmed by another researcher.

To date we have found about twelve million distance-20 positions. The following position was the hardest for our programs to solve:

FU'F2D'BUR'F'LD'R'U'LUB'D2R'FU2D2



6.2 Notes on Cube Explorer

http://kociemba.org/cube.htm to download.
Herbert Kociemba's homepage.

Cube Explorer implements a sophisticated and very powerful algorithm (the Two-Phase-Algorithm). This algorithm provides usually within seconds a solving sequence of 19 moves on average, only one or two moves more than the perfect solution.

Cube Explorer also implements an algorithm to find a perfect solution: a solution which is provably impossible to surpass. With the hardware of year 2010 PC it usually takes only a couple of minutes to find the optimal solution.

Cube Explorer also assists you in finding Pretty Patterns (and provides very short generating maneuvers to carry them out). Cube Explorer searches the universe of Cubes producing a list of all possible Cubes that match your pattern.

In addition to the standard moves of the faces, Cube Explorer also understands slice moves E, M, S and moves of the whole cube x, y and z. Press the associated buttons in the Facelet Editor to understand the meaning of these moves. E', M', S', x', y' and z' are the moves into the opposite direction.

6.2.1 Cube Explorer's algorithm

Herbert Kociemba (HK) developed his two-phase algorithm in 1992. Important ideas from different contributors were added in 2001. A brief description of the algorithm can be found i Cube Explorer's help pages. See chapter "two-phase algorithm".

First phase of solution algorithm uses the following:
If you turn the faces of a solved cube and do not use the moves R, R', L, L', F, F', B and B' you will only generate a subset of all possible cubes. This subset is denoted by G1 = <U,D,R2,L2,F2,B2>. In this subset, the orientations of the corners and edges cannot be changed. That is, the orientation of an edge or corner at a certain location is always the same. And the four edges in the UD-slice (between the U-face and D-face) stay isolated in that slice.

6.3 Popular methods to solve Rubik's Cube

There are many methods to solve the cube. Down below are some of them. Solving by a human or by a computer probably requires two different strategies.

6.3.1 First Corners then Edges

Uses minimum number of simple move sequences to remember.
This method uses very few sequences that you need to memorize in order to solve the cube. Although there are quite a few sequences provided in this solution, most of them are intuitive steps, which once you understand you will never forget. But just like in any other game, you will need to study different methods in order to find your own solving strategy. Even though you might find our corners-first solution pretty straightforward, it will take a lot of practice before you fully master the method.

The method is split to two main steps:

Solve all eight corners of the cube

Solve all twelve edges (and centers) while keeping the corners solved

You might have met other solution method, particularly the most common vanilla "layer by layer" method. This method is an alternative with many advantages:

Smaller amount of unintuitive sequences (nowadays often computer-generated)
Generally shorter sequences
Higher "symmetry" of solution (you solve cube evenly)
You do not break what you solved in previous steps as much (easier recovery from

mistakes)
It is quite efficient with respect to its simplicity
You can achieve very good times just by practice using the very basic method.
You can scale up the method incrementally to gain speed and efficiency

Of course, every single brain works in a bit different way, so what may be an advantage for one may be a disadvantage for others.

The solution is described more in detail in later chapter.

6.3.2 Layer by layer

This method solves all bottom layer cubies, with correct orientation. Then the same for the middle layer and finally the same for the top layer. The solution is described more in detail in later chapter.

6.3.3 Optimized Beginner's Method

Suitable to speed up the simple method with minimum additional effort and possibility to gradually extend to an expert method. More info at http://rubikscube.info/.

6.3.4 Ortega's Method

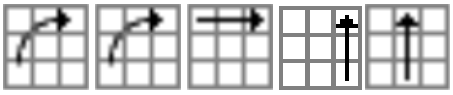
Optimized to the reasonable number of sequences, quick recognition, and turn efficiency.
http://rubikscube.info/ortega.php

6.3.5 Waterman's Method

One of the most complex and optimized methods, many sequences to learn.
Nice Java animation to show how-to: http://rubikscube.info/waterman/stage1.htm

7 Detailed cube solutions (some of them)

7.1 Graphical Notation

 = F2U'RM'

7.2 First Corners then Edges

7.2.1 Corners

Solve Four Bottom Corners

We will start by solving four corners of the cube that share one color (in this case we will select white color). You may try to complete this step on your own and look at the provided sequences only if stuck and frustrated for some time. This step can be solved intuitively if you invest some of your time.

Select one corner with white sticker and turn the whole cube so the white sticker of this corner is facing down. You have solved one of 4 corners this way. Now look for other corners with white sticker and put them to the bottom layer using the right one of the following sequences. Solve the corners one by one. When searching for the next corner to solve, you may freely turn the top layer to put the corners into the position in which you can apply the sequence.

Pay attention to align colors of the corners on sides, since if they do not match as well, the corners are not in correct places. The orange and green colors are just example, there can be other color combinations (like blue-red, green-red, ...) in the pictures instead, just the white sticker should be really white.

The cubie on top, bottom sticker on the front side:

  = F'U'F

The cubie on top, bottom sticker on the right side:

  = RUF'

If you do not see any situation being similar to one of the first two above (remember that you can freely turn top layer to position the corner to the top-right-front position), the corners are in positions that are more difficult to solve.

The following sequences will help you to transform such positions into the ones above you should be familiar with already.

The bottom sticker on the top:

  = RU'R'U2

The cubie on bottom, bottom sticker on the front side:

  = RU'R'

The cubie on bottom, bottom sticker on the right side:

  = F'UF'

One possible way to remember the last two sequences is "bring white sticker to the top, put it back (inside layer you just turned), reverse the first step".

If you have no idea, what is going on when using these sequences, just go really slow and watch what is happening with the solved corners and the one being solved after each turn.

Place Four Top Corners

To solve the four top corners you will **need to temporarily destroy the 4 bottom corners**. The question is: How to destroy and restore the bottom corners so as the top corners become solved? The simplest idea is to remove one bottom corner from its position (using one of the sequences given earlier) and solve it in a different way. Let us look at an example showing the removing and restoring of the front-right-bottom corner:

Remove, position top layer, and restore corner (shown applied to a solved cube):

   = RUR'U'F'U'F

If you look at the result you may notice that the top corners changed. Two corners are twisted (orange-blue-yellow and red-blue-yellow) and two are swapped (top-right ones).
- If we select carefully how to turn the whole cube before applying this *corner* sequence to affect the right corners, we can solve the top corners just by this one sequence!
In this step, we will only position the corners to their correct positions while ignoring the way how they are twisted. Thus our task is quite simple: apply the corner sequence

(possibly more times) to place the corners to their correct positions (use the colors of side stickers of bottom corners to find the right ones).

As you can notice the corner sequence swaps the top-right-front and top-right-back corners. You just need to turn the top-layer and/or the whole cube (keeping top layer facing up) to a position where swapping these two top-right corners will place at least one corner to a correct position. You can always get one of the following cases when turning the top layer to place the corners:

All corners are in their positions (although probably twisted) - this step is finished.

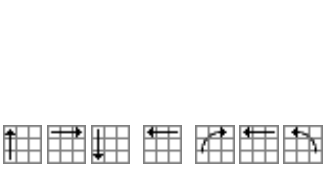
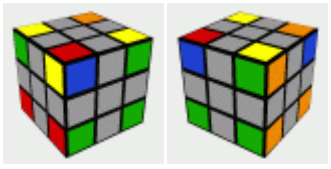
If two adjacent corners can be correctly positioned by turning the top face then only one swap of the other two corners is necessary (make sure that you turn the cube so that these two corners are in top-right positions when applying the sequence).

If two (diagonally) opposite corners can be correctly positioned by turning the top face then perform a swap of any two top corners and you will obtain the previous situation.

Twist Four Top Corners, but not move them

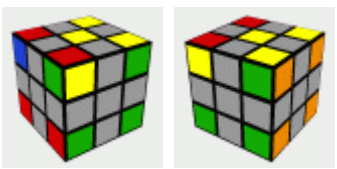
Now we are able to position the top corners using one (quite simple) corner sequence that is explained in the previous text, thus there is no magic here up to this point. Let us try to follow this way even for twisting the corners. We can twist (two) corners using the previous corner sequence, however, it also moves corners which is not good for this step. (Just reminding you that **in this step, we want to twist corners and NOT move them**, because they are already positioned in` the previous step.) The idea behind the corner sequence was to do some change and redo it in a different way, so the other parts of the cube became changed while everything solved before remains solved. Let us try the same idea in this step using our corner sequence: swap two corners using the corner sequence and swap them back from a different angle using the same corner sequence. If we can do so, the corners will be in their correct positions (swap + swap back = nothing), but will be somehow twisted. Let us try that, but before that I must say that swapping two corners back from a different angle requires *left* | *right* mirroring the corner sequence, which is shown below:

Mirror version of the corner sequence (shown applied to a solved cube):



= L'FR'UFUF'

Now you can try the presented idea of doing and redoing (in a different way) the corner swap to twist top corners:

Normal corner sequence, turn cube, mirrored corner sequence (shown applied to a solved cube):



= RUR'U'F'U'FyyL'U'LUFUF'yy (bottom corners unchanged, upper: two twisted)

You can see that this new *twist* sequence leaves all corners in their original positions and **twists only two corners**: the top-left-front corner is twisted clockwise and the top-left-back corner counter-clockwise.
It is not difficult to twist all top corners in any orientation using this twist sequence. After performing consequent multiple application of the twist sequence to the corners, as soon as three corners become oriented correctly, the remaining corner has no chance to be twisted incorrectly (if you do not believe me, try it).

7.2.2 Edges

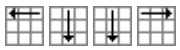
7.2.2.1 Solve Three Left Edges, "Ledges"

To solve the Ledges (which stands here for left-side edges - those with white stickers in the pictures) you use the following simple sequences.  

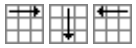
Ledge in bottom-front:

   = U'MU

Ledge in front-bottom:

   = UM'M'U'

Ledge in top-right:

   = U'M'U

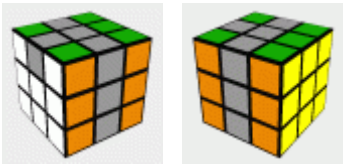
Ledge in right-top:

   = UM'UUMU

Ledge in top-left (flipped in its place):

   = U'MUUMMU

7.2.2.2 Solve Four Redges:



Redges are right-side edges, which have yellow stickers in the pictures.

Redge in bottom-front:

= UMU'

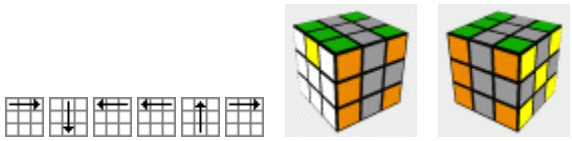
Redge in front-bottom:

= U'M'M'U

Redge in top-left:

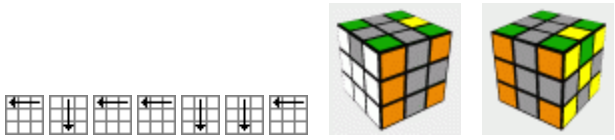
= UM'U'

Redge in left-top:



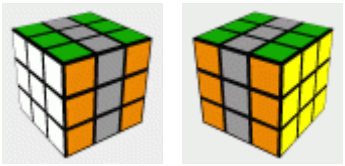
= U'M'UUMU'

Redge in top-right (flipped in its place):

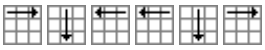


= UM'UUM'M'U

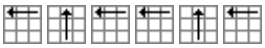
7.2.3 Solve Last Ledge



Ledge in bottom-front:

   = U'M'UUM'U'

Ledge in bottom-back:

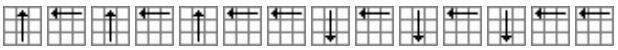
   = UMUUMU

7.2.3.1 Flip Midges (middle edges)

The edges in the ring needs to be flipped in most cases before you can proceed to the following step of positioning them. How should you know which ones need to be flipped? There is a simple rule to spot the incorrectly oriented edges: Look at two colors - color of an edge sticker (choose either one of the two) and color of the center adjacent to the chosen edge sticker. If the colors are the same or opposite (red + orange or blue + green) the edge is just fine. It is flipped otherwise. There may be only none, two, or four of them.

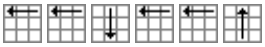
Two top midges flipped:

 = MUMUMUUM'UM'UM'UU

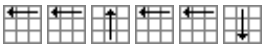
This sequence is so symmetrical and easy to remember that it is hard to forget after you learn it (rumor has it that Rubik himself found this sequence).

7.2.3.2 Place Midges

Three midges in forward cycle:

   = UUM'UUM

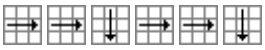
Three midges in backward cycle:

   = UUMUUM'

Two top midges and two bottom midges swapped:

   = F'F'M'M'F'F'M'M'

Two and two midges diagonally swapped:

   = EEM'EEM'

7.3 Layer-by-layer, the 'vanilla' method

7.3.1 Solving the first, upper layer

Solve the first layer as the "upper" layer by positioning the Cube that way.

7.3.1.1 First layer Edge cubies, with correct orientation

This is easy, problem might be the change of orientation of a cubie
RU'BU "changes orientation of ur cubie"
L'UB'U' "changes orientation of ul cubie"
FU'RU "changes orientation of uf cubie"
BU'LU "changes orientation of ub cubie"

7.3.1.2 First layer Corner cubies, with correct orientation

front face only "fru cubie"
FD'F'R'DDR "cycles fld cubie to frd cubie,
frd cubie to fru cubie,
fru cubie to fld cubie,
fru cubie orientation is changed"
(FD'F'R'DDR)^2 "moves fld cubie to flu cubie"
"front-face, diagonal, lower left to upper right"
(FD'F'R'DDR)^3 "turns the fru cubie "
(FD'F'R'DDR)^6 " "
(FD'F'R'DDR)^9 " " " this goes through all orientations of cubie
left face only "lfu cubie"
LD'L'F'DDF , ^2, ^3, ^6, ^9 " the same as above, but for lfu cubie"
back face only "blu cubie"
BD'B'L'DDL , ^2, ^3 ^6 ^9 " the same as above, but for blu cubie"
right face only "rbu cubie"
RD'R'B'DDB , ^2, ^3 ^6 ^9 " the same as above, but for rbu cubie"
moving between faces
" frd -> urf move type: FDF' "
FDF' "cycles fru cubie, to brd, to bld, to frd, back to fru,

	when back to fru it has the same orientation"
LDL'	"cycles lfu cubie, to frd, to rbd, to lfd, back to lfu,
	when back to lfu it has the same orientation"
BDB'	"cycles blu cubie, to fld, to rfd, to bld, back to blu,
	when back to blu it has the same orientation"
RDR'	"cycles rbu cubie, to bld, to rfd, to fld, back to rbu,
	when back to rbu it has the same orientation"

" frd -> fur move type: F'D'F "

R'D'R	"cycles rfu cubie, to fld, to brd, to rfd, back to rfu,
	when back to rfu it has the same orientation"
B'D'B	"cycles bru cubie, to frd, to fld, to brd, back to bru,
	when back to fru it has the same orientation"
L'D'L	"cycles lbu cubie, to brd, to frd, to lbd, back to bru,
	when back to fru it has the same orientation"
F'D'F	"cycles flu cubie, to bld, to brd, to fld, back to flu,
	when back to fru it has the same orientation"

7.3.1.3 Middle layer

Get the edge cubies of the middle layer in place, with correct orientation

(RRUU)^3 = RRUURRUURRUU switches uf and ub, and rf and rb, it is its own inverse

To move an edge cubie from the top layer to the middle layer

FURRUURRUURRUUU'F' = FU(RRUU)^3U'F'

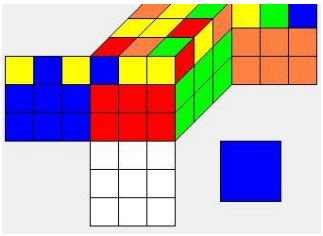
7.3.1.4 Last layer

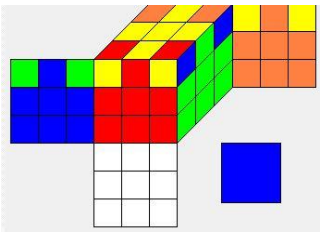
<u>Get a cross at last layer</u>

state Of Cube:

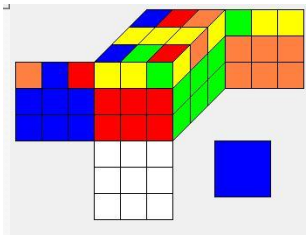
If a cross ... go to next step

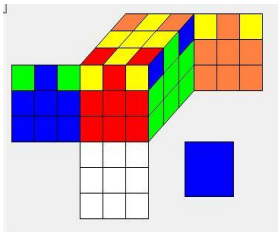
If backward L, use FURU'R'F'





If a line, use FRUR'U'F'





If only centerpiece is correct, this is a combination of the two above, so use those moves

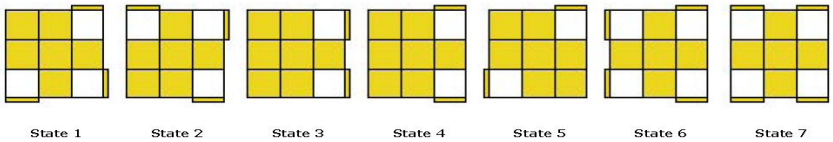
When you have the last layer cross ...

Permuting last layer corners

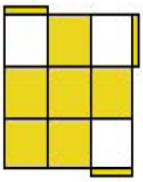
- Swapping adjacent corners, fru <-> bru, use LU'R'UL'U'RUU Keeps the cross but changes order of edgies.
- Swapping diagonal corners and twice swapping adjacent corners, use URU'L'UR'U'L This move changes only the corner cubies.

Orienting the last layer corners

There are 8 possible states for the last layer corners. One is where the corners are correctly oriented.
The other 7 look like this :



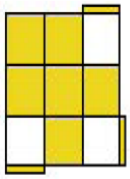
state 1-2 :



State 2. Twisting three corners clockwise

R U R' U R U2 R' U2

= R'U'RU'R'U2RU2



State 1. Twisting three corners anti-clockwise

R' U' R U' R' U2 R U2

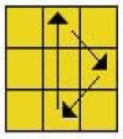
= RUR'URU2R'U2

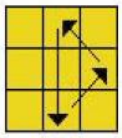
state 3-7:

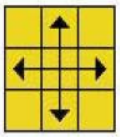
Once you know the algorithms for States 1 and 2, you can solve any LL orientation State
The remaining States can be oriented using a maximum of 2 algorithms.
You will need to do one of the following:
(i) the State 1 algorithm twice,
(ii) the State 2 algorithm twice,
(iii) the State 1 algorithm, then the State 2 algorithm,
or (iv) the State 2 algorithm, then the State 1 algorithm.

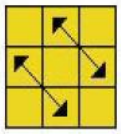
Permuting the last layer edges

4 possible states :


State 1


State 2


State 3


State 4

R2UFB'R2F'BUR2 R2U'FB'R2F'BU'R2

State 3 and 4:
Do either state 1 or state 2 move => state 3 and 4 will be state 1 or 2

That's it, layer-by-layer. Not as easy as one would like, but all methods needs remebering a number of moves, and in the worst case 20 moves, so the lord has spoken :-)

8 References

[W.D.J] Mathematics of the Rubik's cube, Professor W.D. Joyner 1996-97
http://www.usna.edu/Users/math/wdj/rubik_nts.htm and
http://www.permutationpuzzles.org/rubik/webnotes/rubik.pdf

[J.C.] Group Theory and the Rubik's Cube, Janet Chen
http://www.math.harvard.edu/~jjchen/docs/
Group%20Theory%20and%20the%20Rubik%27s%20Cube.pdf

[Si] D. Singmaster, Notes on Rubik's Magic Cube, Enslow, 1981
Can be bought used from amazon.co.uk

[CL] Cube lover's list, http://www.math.ucf.edu/~reid/Rubik/index.html

[BBBC] Group Theory, B Baumslag, B Chandler
ISBN 13:978-0070041240 ISBN 10:0070041245

[TW] A. D. Thomas and G. V. Wood, Group Tables,
Shiva Publishing Ltd, Kent, UK, 1980

[REK] Richard E. Korf Finding Optimal Solutions to Rubik's Cube Using Pattern Databases
http://www.cs.princeton.edu/courses/archive/fall06/cos402/papers/korfrubik.pdf
or http://www-compsci.swan.ac.uk/~csphil/CS335/korfrubik.pdf

[GAP] http://www.gap-system.org/ GAP - Groups, Algorithms, Programming - a System for Computational Discrete Algebra, emphasis on computational group theory.

[Cube Explorer] http://kociemba.org/cube.htm

[Atlas] Atlas of finite groups: http://web.mat.bham.ac.uk/atlas/v1.html and
http://brauer.maths.qmul.ac.uk/Atlas/v3/

Various material from Wikipedia.org, rubikscube.info, cube20.org,
http://en.wikibooks.org/wiki/How_to_Solve_the_Rubik%27s_Cube,

9 Appendix, Generators and relations, |G| < 26

C_n = cyclic group of order n,
D_n = dihedral group of order 2n = symmetry group of the regular n-gon,
Q = quaternion group = {-1, 1, -i, i, -j, j, -k, k},
S_n = symmetric group of permutations of {1, 2, ... n},
A_n = alternating group of even permutations of {1, 2, ... n},
F_q = finite field with q elements (q=power of a prime),
Z/n**Z** = integers modulo n.
(Tables from [W.D.J] and more information can be found in [Atlas])

Order	Group G	generators	relations	notes
2	C_2	a	$a^2=1$	
3	C_3	a	$a^3=1$	$G=A_3$
4	C_4	a	$a^4=1$	
4	$C_2\times C_2$	a,b	$a^2=1, b^2=1,$ $ab=ba$	Klein 4-group $Aut(G)=GL(2,\mathbb{F}_2)$
5	C_5	a	$a^5=1$	
6	$C_6=C_2\times C_3$	a	$a^6=1$	
6	S_3	a,b	$a^3=1, b^2=1,$ $aba=b$	$Aut(G)=G$ $G=GL(2,\mathbb{F}_2)$
7	C_7	a	$a^7=1$	
8	C_8	a	$a^8=1$	
8	$C_2\times C_4$	a,b	$a^2=1, b^4=1,$	
			$ab=ba$	
8	$C_2\times C_2\times C_2$	a,b,c	$a^2=1, b^2=1,$ $c^2=1, ab=ba,$ $bc=cb, ac=ca$	$Aut(G)=GL(3,\mathbb{F}_2)$
8	D_4	a,b	$a^4=1, b^2=1,$ $aba=b$	
8	Q	a,b	$a^4=1, b^2=a^2,$ $aba=b$	
9	C_9	a	$a^9=1$	
9	$C_3\times C_3$	a,b	$a^3=1, b^3=1,$ $ab=ba$	$Aut(G)=GL(2,\mathbb{F}_3)$
10	$C_{10}=C_2\times C_5$	a	$a^{10}=1$	
10	D_5	a,b	$a^5=1, b^2=1,$ $aba=b$	
11	C_{11}	a	$a^{11}=1$	
12	$C_{12}=C_3\times C_4$	a	$a^{12}=1$	
12	$C_2\times C_6$ $=C_2\times C_2\times C_3$	a,b	$a^2=1, b^6=1,$ $ab=ba$	
12	D_6	a,b	$a^6=1, b^2=1,$ $aba=b$	
12	A_4	a,b	$a^2=1, b^3=1,$ $(ba)^3=1$	$Aut(G)=G$
12	Q_6	a,b	$a^6=1, b^2=a^3,$ $aba=b$	"dicyclic"

13	C_{13}	a	$a^{13}=1$	
14	$C_{14}=C_2\times C_7$	a	$a^{14}=1$	
14	D_7	a,b	$a^7=1, b^2=1,$ $aba=b$	$Aut(G)=G$
15	$C_{15}=C_3\times C_5$	a	$a^{15}=1$	
16	C_{16}	a	$a^{16}=1$	
16	$C_2\times C_8$	a,b	$a^2=1, b^8=1,$ $ab=ba$	
16	$C_4\times C_4$	a,b	$a^4=1, b^4=1,$ $ab=ba$	$Aut(G)=GL(2,\mathbb{Z}/4\mathbb{Z})$
16	$C_2^2\times C_4$	a,b,c	$a^2=1, b^2=1, c^2=1,$ $ab=ba, ac=ca, bc=cb$	
16	C_2^4	a,b,c,d	$a^2=1, b^2=1,$ $c^2=1, d^2=1,$ $ab=ba, ac=ca, ad=da,$ $bc=cb, bd=db, cd=dc$	$Aut(G)=GL(4,\mathbb{F}_2)$
16	$D_4\times C_2$	a,b,c	$a^4=1, b^2=1, c^2=1,$ $aba=b, ac=ca, bc=cb$	
16	$Q\times C_2$	a,b,c	Exercise	
16		a,b,c	$a^2=1, b^2=1, c^2=1,$ $abc=bca=cab$	
16		a,b	$a^2=1, b^2=1,$ $(ab)^2=1, (a^{-1}b)^2=1$	
16		a,b	$a^4=1, b^4=1,$ $aba=b$	a semidirect product of C_4 with C_4
16		a,b	$a^8=1, b^2=1,$ $ab=ba^5$	a semidirect product of C_8 with C_2 (C_2 normal)
16		a,b	$a^8=1, b^2=1,$ $ab=ba^3$	a semidirect product of C_8 with C_2 (C_2 normal)
16	D_8	a,b	$a^8=1, b^2=1,$ $aba=b$	a semidirect product of C_8 with C_2 (C_2 normal)
16	Q_8	a,b	$a^8=1, b^2=a^4,$ $aba=b$	
17	C_{17}	a	$a^{17}=1$	
18	$C_{18}=C_2\times C_9$	a	$a^{18}=1$	

18	$C_3 \rtimes C_6$ $= C_3 \times C_3 \rtimes C_2$	a, b	$a^3 = 1, b^6 = 1,$ $ab = ba$	
18	$S_3 \times C_3$	a, b, c	$a^3 = 1, b^2 = 1, c^3 = 1$ $aba = b, ac = ca, bc = cb$	
18	D_9	a, b	$a^9 = 1, b^2 = 1,$ $aba = b$	a semidirect product of C_9 with C_2 (C_2 normal), $Aut(G) = G$
18		a, b, c	$a^3 = 1, b^3 = 1, c^2 = 1$ $ab = ba, aca = c, bcb = c$	
19	C_{19}	a	$a^{19} = 1$	
20	$C_{20} = C_4 \times C_5$	a	$a^{20} = 1$	
20	$C_2 \times C_{10}$	a, b	$a^2 = 1, b^{10} = 1,$ $ab = ba$	
20	D_{10}	a, b	$a^{10} = 1, b^2 = 1,$ $aba = b$	
20	Q_{10}	a, b	$a^{10} = 1, b^2 = a^5,$ $aba = b$	
20		a, b	$a^5 = 1, b^4 = 1,$ $ab = ba^3$	a semidirect product of C_5 with C_4 (C_4 normal), $Aut(G) = G$
21	$C_{21} = C_3 \times C_7$	a	$a^{21} = 1$	
21		a, b	$a^7 = 1, b^3 = 1,$ $ab = ba^4$	a semidirect product of C_7 with C_3 (C_3 normal), $Aut(G) = G$
22	$C_{22} = C_2 \times C_{11}$	a	$a^{22} = 1$	
22	D_{11}	a, b	$a^{11} = 1, b^2 = 1,$ $aba = b$	$Aut(G) = G$
23	C_{23}	a	$a^{23} = 1$	
24	$C_{24} = C_3 \times C_8$	a	$a^{24} = 1$	
24	$C_2 \times C_{12}$ $= C_2 \times C_3 \times C_4$	a, b	$a^2 = 1, b^{12} = 1,$ $ab = ba$	
24	$C_2^2 \times C_6$	a, b, c	$a^2 = 1, b^2 = 1, c^6 = 1,$ $ab = ba, ac = ca, bc = cb$	

24	$D_6 \times C_2$	a, b, c	$a^6 = 1, b^2 = 1, c^2 = 1,$ $aba = b, ac = ca, bc = cb$	
24	$A_4 \times C_2$	a, b, c	Exercise	
24	$Q_6 \times C_2$	a, b, c	Exercise	
24	$D_4 \times C_3$	a, b, c	Exercise	
24	$Q \times C_3$	a, b, c	Exercise	
24	$S_3 \times C_4$	a, b, c	Exercise	
24	D_{12}	a, b	$a^{12} = 1, b^2 = 1,$ $aba = b$	
24	Q_{12}	a, b	$a^{12} = 1, b^2 = a^6,$ $aba = b$	
24	S_4	a, b	$a^4 = 1, b^2 = 1,$ $(ab)^3 = 1$	$Aut(G) = G$
24	$SL(2, \mathbb{F}_3)$	a, b, c	$a^4 = 1, b^2 = a^2,$ $c^3 = 1, aba = b,$ $ac = cb, bc = cab$	$Aut(G) = Aut(Q)$ $= S_4,$ a semidirect product of Q with C_3 (C_3 normal)
24		a, b	$a^3 = 1, b^8 = 1,$ $aba = b$	a semidirect product of C_3 with C_8 (C_8 normal)
24		a, b, c	$a^3 = 1, b^4 = 1,$ $c^2 = 1, bcb = c, aba = b,$ $ac = ca$	a semidirect product of C_3 with D_4 (D_4 normal)
25	C_{25}	a	$a^{25} = 1$	
25	C_5^2	a, b	Exercise	