

5. THE TODD-COXETER ALGORITHM

§5.1. Presentations

Presenting a group by generators and relations is by far the most compact way of describing a group and it is often the way it arises in applications. So we need some way of unravelling the structure of the group from its presentation. The Todd-Coxeter Algorithm aims to do just that.

If Γ is a set of symbols, a **group word** on Γ is a string of elements of Γ and their inverses. The inverse of a symbol X is just the formal expression X^{-1} , treated as a single symbol. For convenience in our calculations we will use lower case letters to denote inverses. So, for example, we use the letter a to represent A^{-1} .

Example 1: The group word $AAbABAaa$ represents $A^2B^{-1}ABA^{-3}$. If $A^4 = 1$, the group words $AAAAA$, A and aaa all represent the same element.

In the following description of the algorithm we sometimes use a capital letter, such as X , to represent either a generator or the inverse of a generator. In that case the corresponding lower case letter represents its inverse. So if X is actually the inverse of a generator, x would be the corresponding generator itself.

A **presentation** of a group is a description $\langle \Gamma \mid R \rangle$ where Γ is a set of **generators** and R a set of group words, called **relators**. Every element of the group is represented by a group word on Γ . Two group words represent the same element if the equality is a consequence of the relations in R . (In chapter 9 we'll define $\langle \Gamma \mid R \rangle$ more precisely as a quotient group of a free group.) Sometimes we write a relator $U^{-1}V$ or UV^{-1} as the **relation** $U = V$.

A **free group** is one with no relations, that is, G is free if $G \cong \langle \Gamma \mid \rangle$ for some Γ . A free group on a single generator is the infinite cyclic group.

A group G is **finitely generated** if it has a presentation $\langle \Gamma \mid R \rangle$ where Γ is finite and it is **finitely presented** if it has a presentation $\langle \Gamma \mid R \rangle$ where both Γ and R are finite. In this chapter we only consider finitely presented groups.

Example 2: $\langle A, B \mid A^4, B^2, BA = A^{-1}B \rangle$ is a **finitely presented group**.

We will write this as $\langle A, B \mid A^4, B^2, BA = aB \rangle$.

We can also write this as $\langle A, B \mid A^4, B^2, ABAB \rangle$ because we can deduce the relation $(AB)^2 = 1$ from $A^4 = 1$, $B^2 = 1$ and $BA = A^{-1}B$ and conversely we can deduce $BA = A^{-1}B$ from the relations $A^4 = 1$, $B^2 = 1$ and $(AB)^2 = 1$.

An equivalent presentation of this group is

$\langle A, B \mid A^4, B^2, B^{-1}AB = A^{-1} \rangle$

which we shall write as $\langle A, B \mid A^4, B^2, bAB = a \rangle$.

The elements of this group are products of powers of A's and B's. Since $BA = A^{-1}B$ we can bring all the A's to the front and so every element can be put in the form A^rB^s .

Since $A^4 = 1$ and $B^2 = 1$ we can assume that $r = 0, 1, 2$ or 3 and $s = 0$ or 1 . The group thus has 8 elements: 1, A, A^2 , A^3 , B, AB, A^2B , A^3B . It is the dihedral group of order 8.

You may feel that you were able to predict that this group has order 8 from the relators A^4 and B^2 (since $4 \times 2 = 8$). But it's not always so simple. For example the following group may appear to have order 4 but in fact it is infinite.

Example 3: $\langle A, B \mid A^2, B^2 \rangle$ is infinite.

The elements are strings of alternating A's and B's (any successive pair of A's or of B's can be removed) so the distinct elements are:

1, A, AB, ABA, ABAB, ..., B, BA, BAB, BABA,

For example the product of BAB and BABABA is

$$BABBABABA = ABA.$$

The inverse of BABA is ABAB and the inverse of ABA is ABA itself.

Any string that starts and finishes with the same symbol has infinite order while if the first symbol is different to the last the string has order 2.

So $C = AB$ has infinite order as has $C^{-1} = B^{-1}A^{-1} = BA$. The group can be generated by A and C (since B =

AC) and since $BC = BAB = C^{-1}B$ we can present the group as $\langle C, B \mid B^2, BC = C^{-1}B \rangle$. Rewriting C as A this becomes $\langle A, B \mid B^2, BA = A^{-1}B \rangle$ which we recognise as belonging to the dihedral family. We call it the **infinite dihedral group** and denote it by \mathbf{D}_∞ .

On the other hand the group in the next example is much smaller than it might appear.

Example 4: $\langle A, B \mid A^5, B^3, BA = A^2B \rangle$ has order 3. This is because $BAB^{-1} = A^2$ so

$$\begin{aligned} B^2AB^{-2} &= B(BAB^{-1})B^{-1} \\ &= BA^2B^{-1} \\ &= (BAB^{-1})^2 \\ &= (A^2)^2 \\ &= A^4. \end{aligned}$$

Continuing we get $B^3AB^{-3} = A^8 = A^3$.

Since $B^3 = 1$ it follows that $A = A^3$ and hence $A^2 = 1$.

But $A^5 = 1$ so $A = 1$. So this is simply the cyclic group of order 3 in disguise.

A simpler presentation would be $\langle B \mid B^3 \rangle$ or, since we could use any letter for the generator, $\langle A \mid A^3 \rangle$.

§5.2. Chains and Links

Suppose we have a finite presentation $\langle \Gamma \mid R \rangle$ for a group G . An element of G will be represented by many group words. We attempt to assign to each element a unique integer code, with 1 representing the identity and 2, 3, ...,

N representing the other elements. If we succeed, the order of the group is N . Of course this will only happen if G is finite.

A **chain** is an expression mWn , representing the equation $mW = n$, where m and n are integer codes and W is a group word.

The **length** of a chain mWn is the length of the word W (that is, the number of symbols, counting each generator or its inverse as a single symbol).

Example 5: $7AAbbb2$ is a chain of length 5 representing $7 \times A^2B^{-3} = 2$. Here 2, 7 and A^2B^{-3} are all elements of the group. The elements represented by the integer codes 2 and 7 will also be expressible as group words and the group word A^2B^{-3} will also have an integer code.



At the outset the elements of the group being presented will be expressed as words in the generators but, since many different words can represent the same element, this is not a satisfactory notation.

Gradually, as the algorithm progresses, we assign a unique integer code to each element.

A chain mWn is a **link** if it has length 1, that is, where the word W is a generator or the inverse of a

generator. Once we know all the links we can express the codes in terms of the generators and hence obtain the group table.

Example 6: If B is one of the generators then the chain $6B2$ is a link. It provides the information that $6B = 2$. For example, if we have already expressed 6 as the word BA^3B^{-2} then we can express 2 as BA^3B^{-1} .

The links are used to build up a table, called the **Link Table**:

	A	B	...
1			
2			
...			

Once we have completed this table we can produce the group table.

Example 7: If we have a group generated by A and B with the following link table, the group has order 6.

	A	B
1	2	4
2	3	5
3	1	6
4	6	1
5	4	2
6	5	3

We can use the link table to express each element as a group word. For example, since 1 is the identity it follows that the element 2 is A and the element 4 is B. In the next row we have the fact that $2A = 3$ so $3 = AA = A^2$. Doing this for all the elements we get:

	A	B	
1	2	4	1
2	3	5	A
3	1	6	$2A$
4	6	1	B
5	4	2	$2B$
6	5	3	$3B$

The group word associated with each is, of course, not unique. We can now complete the group table.

For example, $5.3 = 5.2A = 5AA = 4A = 6$ and

$$6.5 = 6.2B = 6AB = 5B = 2.$$

If G is a generator then the links rGs and sgr , where $g = G^{-1}$, give equivalent information. This is because rGs represents the equation $rG = s$ while sgr represents the equivalent equation $sG^{-1} = r$.

We call these links **conjugates** of one another.

Example 8: The conjugate of the link 3B5 is the link 5b3.

§ 5.3. Restricted Todd-Coxeter Algorithm

To make it easy to follow the algorithm I'll use a mixture of metaphors. So far I've used the rather mechanical metaphor of 'chains' and 'links' to describe

the ingredients of the algorithm, but to describe the overall behaviour I'll use the metaphor of population dynamics.

A chain is said to **die** when it contracts down to a link. Throughout the course of the algorithm chains are born, contract and eventually die and the success of the algorithm for a particular presentation depends on the balance between the birth and death rates. The birth rate is uniform (with as many new chains born in each generation as there are relators) but the death rate can be unpredictable. At first the set of live chains grows, with more being born than die. If all goes well the time comes when the balance shifts and the population decreases and is finally wiped out. But it can happen that the population grows indefinitely and in this case the algorithm will fail to terminate.

The algorithm proceeds in a cycle with three stages occurring at each generation:

BIRTH, CONTRACTION and MARRIAGE.

We begin with **BIRTH**.

BIRTH RULE

Having just created the integer code m we add the chain mRm to the list of chains for each relator R .

Suppose we are processing the presentation
 $\langle A, B, \dots | R, S, \dots \rangle$.

Then the relators R, S, \dots represent the identity. So for every integer code m , the chains mRm, mSm, \dots are valid because they simply state that $m1 = m$. At each generation, after introducing a new integer code, these corresponding chains are born, one for each relator.



This is how the algorithm gets off the ground because we begin with the code 1, representing the identity and so the first chains to be born are:

1R1, 1S1,

Example 9: For $\langle A, B \mid A^4, B^2, BA = A^{-1}B \rangle = \langle A, B \mid A^4, B^2, ABAb \rangle$ the process begins with the birth of the chains: 1AAAA1, 1BB1, 1ABAb1.

Next we carry out any contractions that are possible.

CONTRACTION RULE

If X is a generator (or the inverse of a generator) and rXs is a link we may contract rX to s at the start of any live chain and contract X 's to r at the end of any live chain:

$$rX \rightarrow s, \quad Xs \rightarrow r$$

After the first generation of chains are born there are no links and so no contraction is possible. One way to obtain links is for chains to contract to a link, but clearly that cannot occur until we have some links already. Fortunately there's



another way of obtaining links, through so-called 'marriage'.

The link rXs represents the equation $rX = s$, so clearly rX can be replaced by s . If αXs is any live chain, ending in Xs then $s = \alpha X = rX$, so $\alpha = r$. Hence α r is a valid chain (representing $\alpha = r$).

Example 10: If $3A5$ is a link (equivalent to the conjugate link $5a3$) we may perform the following contractions:

$$\begin{aligned}3\mathbf{A}BBB3 &\rightarrow 5\mathbf{B}BB3, \\5\mathbf{a}BBAbA2 &\rightarrow 3BBAbA2, \\2ABA\mathbf{A}5 &\rightarrow 2ABA\mathbf{3}, \\2ABA\mathbf{3} &\rightarrow 2AB5.\end{aligned}$$

If $5B4$, $2B3$ are also links we may make the following further contractions of $5\mathbf{B}BB3$:

$$\mathbf{5}BBB3 \rightarrow \mathbf{4}BB3 \rightarrow 4B2.$$

This chain has now contracted down to a link and so, as a chain, it dies. But it may well be a *new* link, enabling further contractions.

DEATH RULE

**Whenever a chain contracts to a link it ‘dies’.
The information it conveys is transferred to the link table.**

If the new link has the form rGs , where G is a generator, or the inverse of a generator, we write s in row r , column G of the link table. Then we can write the corresponding information for the equivalent link sgr .

When the dust finally settles, and we still have some live chains, but no more contractions are possible, we must create a new integer code.



MARRIAGE RULE

If the set of live chains is non-empty, and no further contractions are possible, assign the next available code, m , by creating a new link of the form mGr or rGm where G is a generator or the inverse of a generator for some $r < m$.

The creation of the new link can be thought of as a ‘marriage’ where the new code ‘marries’ one of the existing ones. Like marriage in western society there is freedom to choose a partner – the algorithm doesn’t arrange the marriage. But, as with real marriages there are sensible choices and less sensible ones, so it is with the Todd-Coxeter Algorithm. At the time of a marriage it is impossible to know for certain that a given choice is going to be successful!

Unlike human marriage the choice goes beyond that of choosing a partner – the choice of generator comes into it too. And different codes can ‘marry’ the same code, using different generators. Perhaps the analogy is starting to break down, so let me explain in a more mundane fashion how this so-called ‘marriage rule’ operates.



The rule defines the next integer code in terms of the preceding ones. Having done that, a whole new generation of chains is ready to be born. Further possibilities for contraction now arise. There is now a whole new generation of chains to process, but more importantly the new link can be used in further contractions.

Example 11: Suppose we have assigned codes 1, 2, 3 and have generated all the corresponding chains and

contracted them as far as possible. At that stage we need to define a new integer code by creating a link to define 4. For example we might create the link 4A2. This means putting $4A = 2$ or, in other words, defining 4 as $2A^{-1}$. Or we could create 3B4, which defines 4 as 3B.

Note that we are not allowed to create a link of the form mAm , defining m in terms of itself. Nor can we define m in terms of a future code. If only codes 1, 2, 3 have been assigned then the code 4 can't be defined by a link such as 4B7. In defining a new code m the other code must be less than m .

One very good strategy is to create the new link so as to fill up the next available blank in the Link Table. This could be considered the **default strategy** and it is the one that is mostly used in the following examples. However the unsolvability of the word problem means that there is no strategy that can be guaranteed to always work. So, there will be times when this default strategy will be the wrong thing to do.

Example 12:

Suppose we have assigned codes 1, 2, 3 and we need to assign the code 4 at the stage where our table of links so far is:

	A	B
1	2	3
2	1	
3		

We may decide to create the link 2B4 or 3A4 or 3B4. The default strategy would be to create the link 2B4.

But another consideration is how useful the newly created link would be at the moment. If, in the above situation, we had the chain 3BA3 we might decide to create the link 3B4 because this could be used immediately to contract 3BA3 to 4A3 thereby giving yet another link.

Finally we must ask, “How does the algorithm begin and how does it terminate?” It begins with the empty set of chains and with the integer code 1 being created, representing the identity. This immediately causes a number of chains to be born, and the process of birth, contraction and death to take place.

STARTING
Begin with an empty set of chains and
assign the code 1.

The restricted algorithm that we are describing terminates in one of two ways. For a start there is the possibility that a chain contracts to a link which conflicts with one that we already have. That is, when we come to enter the new link in the link table we find that this cell is already occupied *with a different code*. Or we may find that a code is repeated in a row.

For example if we have the two links 5B7 and 5B9 then $5B = 7$ and $5B = 9$, meaning that we have

inadvertently issued two different codes for the same element. This is called a **contradiction**. It could mean that you've made a mistake, but it can also arise with no error being made.

The complete axiom arranges for the two codes to be identified, so that the larger code is replaced by the smaller one in all chains and links. Then the usual process continues. We will only consider the Restricted algorithm, whereby we abort if we reach a contradiction. Of course, you should check your working if a contradiction arises. But if the contradiction is really there the only thing only thing to do is start again with different choices, or perhaps rewrite the presentation and start again. In everything that follows, when we refer to the Todd-Coxeter Algorithm we will mean the restricted one.

It is also possible that we have two links with the same codes but different generators. For example you may find that you get the two links $5A7$ and $5C7$. This would mean that having $5A = 7$ and also $5C = 7$. This would mean that $A = C$. Again this is more likely to be due to an error. But it can occur by virtue of the fact that the relators in the presentation might lead to such a collapse. For example in the presentation

$$\langle A, B, C \mid A^4, B^2, (A^{-1}C)^2, (A^{-1}C)^7, BA = A^{-1}B \rangle,$$

the relators $(A^{-1}C)^2$ and $(A^{-1}C)^7$ together imply that $A^{-1}C = 1$, or $A = C$. Such a situation can't be remedied by making better choices. The problem is intrinsic to the presentation. If you ever get such a situation you should

first check your work, and then, if you are convinced you have made no error, modify the presentation.

But the normal way for the algorithm to terminate, and the only way that gives us a group table, is to arrive at a situation where we have a complete Link Table, with no blanks. Clearly a completed link table will allow all chains to be contracted to links, so that all chains will die, but it is not necessary to do this.

FINISHING:

Terminate when either:

- (1) All chains have become links and the table of links is complete or**
- (2) You get a contradiction in the Link Table (an abortion).**

If the algorithm terminates under (1) it will have been successful and we will have a group table. If it terminates under (2) the algorithm fails. This will either mean that we could have made better choices (we could begin again and vary our strategy). A further possibility is that we never reach either state (1) or (2) and the algorithm fails to terminate. This will occur if the group is infinite.

With the Unrestricted Algorithm, where identifications are used to deal with contradictions, it is the case that if the group is finite there will definitely be a sequence of choices that will lead to the algorithm

terminating successfully. The problem is that there is no strategy that can be guaranteed to achieve this.

THE (RESTRICTED) TODD-COXETER ALGORITHM

Suppose $\langle G_1, G_2, \dots, G_m \mid R_1, R_2, \dots, R_n \rangle$ is a presentation for a group G .

- (1) Initially let $N = 1$, C = the empty set of chains, and L a table with $2m$ columns with each cell blank. The number of rows is initially 1. The $2m$ columns are labelled by the generators and their inverses.
- (2) For each relator R_i adjoin the chain NR_iN to C ;
- (3) Where possible, contract chains using the links in L . Any chain that contracts to a link is transferred to the link table and the chain dies. It's removed from the chain table, C .
- (4) If this results in two integers being assigned to the same cell in the Link Table, abort the algorithm.
- (5) When no further contraction is possible, and L has a blank cell, increase N by 1. Then marry N to one of the existing codes by inserting a new link into the Link Table. Then GO TO (2).
- (6) If L has no blank cell terminate.

Then the group has order N and the group table can be constructed from the information given by the links in L .

In performing the Todd-Coxeter Algorithm by hand we set out our working in three tables:

CHAIN TABLE: This describes the successive contractions of the chains. Each of the chains that are born is placed at the start of a new row and their successive contractions are recorded.

LINK TABLE: Once a proper chain becomes a link the information is transferred to the link table. At the right of the table record the link definitions for each code. The code ‘1’ is assigned to the identity so we write 1 in the definition column of the table for the code ‘1’.

	A	B	...	
1			...	1
2			...	
...	

If the relators involve inverse it is useful to have extra columns for inverses of generators. For example, if we have the relator $CBC^{-1}A$, which we would write as $CBcA$, we should have a column for c . Whenever we have a link mCn we should enter n in the mC position of the Link Table and m in the nc position. This is because mCn and the conjugate ncm , carry the same information since mCn means $mC = n$ and ncm means $nC^{-1} = m$.

Suppose we have just created, or discovered, the link $2C3$. Then we enter it, and the conjugate link $3c2$ into the Link Table as follows:

	A	B	C	c	
1					1
2			3		
3				2	

But don't bother with these inverse columns if there are no inverses in the relators.

GROUP TABLE: This is the ‘multiplication’ table for the group.

	1	2	...
1	1	2	...
2	2		...
...

§ 5.4. Examples

Perhaps you find all this very confusing. That's not surprising, because this is probably the most complex algorithm you'll ever have to carry out. Also, what you might find disquieting, is that the algorithm is not deterministic. At one stage in every generation you have to make a ‘marriage’ choice and the success depends on whether you make the ‘right’ one. I can give advice that can help, but the unsolvability of the word problem guarantees that no choice strategy exists that would always be successful.

Having said this, it is remarkable how often even the restricted algorithm works, provided we start with a ‘good’ presentation and use the couple of simple

strategies that I describe. Work through the following examples and you'll eventually find that it becomes second nature.

Example 13: $G = \langle A, B \mid A^2, B^2, (AB)^2 \rangle$.

(For clarity I'll describe the appearance of the tables at each generation and make bold the additional information at each stage. Normally you would carry out your working in the one set of tables.)

We begin with $N = 1$. For each relator a chain is born.

1AA1
1BB1
1ABAB1

	A	B
1		

1

Having no links we clearly can't make any contraction so we must create a link to define 2 (the marriage step). Using the default strategy we create the link 1A2, thereby defining $2 = A$. We write 2 in the '1' row and A column. We enclose it in parentheses to record the fact that this link was created, rather than found as a result of contracting a chain. This makes it easier to follow the sequence of events when the table is completed.

1AA1
1BB1
1ABAB1

	A	B
1	(2)	
2		

1

A

Having defined a new code, a new generation of chains is born, one for each relator.

1AA1
1BB1
1ABAB1
2AA2
2BB2
2ABAB2

	A	B	
1	(2)		1
2			A

Some contraction is now possible. Any ‘1A’ can be replaced by ‘2’ and any ‘A2’ can be replaced by ‘1’. We use an arrow to separate a chain from its contraction.

1AA1→2A1
1BB1
1ABAB1→2BAB1
2AA2→2A1
2BB2
2ABAB2

	A	B	
1	(2)		1
2			A

Three chains have contracted, and two of them have contracted down to the same link 2A1. This is a new link and so is transferred to the link table. Those two chains are considered ‘dead’ and don’t enter into the remaining computation so we mark them by the symbol ‘×’.

1AA1→2A1	×	A	B	1
1BB1		1	(2)	
1ABAB1→2BAB1		2	1	A
2AA2→2A1	=			
2BB2				
2ABAB2				

The new link 2A1 has been obtained twice. The first time it gets recorded as ‘×’ but the second time it gets ‘=’, indicating that the link is repeating one we already have. In general, any chain that becomes equal to one that we have elsewhere in the Chain Table get’s killed immediately and is recorded as ‘=’.

The new link means that chains starting with ‘2A’ or ending with ‘A1’ can be contracted. This allows just one more contraction: $2ABAB2 \rightarrow 1BAB2$.

1AA1→2A1	×	A	B	1
1BB1		1	(2)	
1ABAB1→2BAB1		2	1	A
2AA2→ 2A1	=			
2BB2				
2ABAB2→1BAB2				

No further contraction is possible so we must create a new code. The default strategy would create 1B3. This

seems to be a good idea because it will be useful in contracting the two chains that begin with ‘1B’.

1AA1→2A1	×	A	B	
1BB1→3B1	×	1	(2)	(3)
1ABAB1→2BAB1		2	1	
2AA2→2A1	×	3		
2BB2				
2ABAB2→1BAB2→3AB2				
3AA3				
3BB3→3B1	=			
3ABAB3→3ABA1				

We have now (twice) obtained the link 3B1. We enter this up in the link table. This link allows some further contraction.

1AA1→2A1	×	A	B	
1BB1→3B1	×	1	(2)	(3)
1ABAB1→2BAB1→2BA3		2	1	
2AA2→2A1	×	3		1
2BB2				
2ABAB2→1BAB2→3AB2				
3AA3				
3BB3→3B1	=			
3ABAB3→3ABA1→3AB2				

No further contraction is possible so we create code ‘4’.

1AA1→2A1	×	A	B
1BB1→3B1	×	1	(2)
1ABAB1→2BAB1→2BA3	=	2	(3)
2AA2→2A1	=	3	1
2BB2	=	4	
2ABAB2→1BAB2→3AB2	=		
3AA3	=		
3BB3→3B1	=		
3ABAB3→3ABA1→3AB2	=		
4AA4	=		
4BB4	=		
4ABAB4	=		

Many contractions are now possible, leading to further links, leading to further contractions until all chains become links and the link table becomes complete.

1AA1→2A1	×
1BB1→3B1	×
1ABAB1→2BAB1→2BA3→ 4A3	×
2AA2→2A1	=
2BB2→ 4B2	×
2ABAB2→1BAB2→3AB2→ 4B2	=
3AA3→ 3A4	×
3BB3→3B1	=
3ABAB3→3ABA1→3AB2	=
4AA4→ 3A4	=
4BB4→ 2B4	=
4ABAB4→3BAB4→1AB4→ 2B4	=

	A	B	
1	(2)	(3)	1
2	1	(4)	A
3	4	1	B
4	3	2	2B

We can now complete the group table. The first row and column are easy, since '1' is the identity.

	1	2	3	4
1	1	2	3	4
2	2			
3	3			
4	4			

Now $2 = A$ so the '2' column in the group table is a copy of the A column in the Link Table. Similarly, as $3 = B$, the '3' column in the group table is a copy of the B column in the Link Table.

	1	2	3	4
1	1	2	3	4
2	2	1	4	
3	3	4	1	
4	4	3	2	

Finally, since $4 = 2B$ we use the B column of the Link Table to transform the '2' column of the group table into the '4' column. For example, $3.4 = 3(2B) = (3.2)B = 4B$ (from the '2' column of the group table) = 3 (from the B column of the link table).

The entries in the ‘2’ column are 2, 1, 4, 3 respectively and from the link table $2B = 4$, $1B = 3$, $4B = 2$ and $3B = 1$. So the ‘4’ column of the group table is 4, 3, 2, 1 respectively.

	1	2	3	4
1	1	2	3	4
2	2	1	4	3
3	3	4	1	2
4	4	3	2	1

Example 14: $G = \langle A, B \mid A^3, B^2, BAB = A^{-1} \rangle$

We write the third relator as $BABA$. We show the state of the computation at each generation and, to save space, we omit any rows that contain ‘dead’ chains from the previous step.

N = 1, 2

1AAA1 \rightarrow 2AA1
1BB1 \rightarrow
1BABA1
2AAA2 \rightarrow 2AA1
2BB2
2BABA2 \rightarrow 2BAB1

	A	B	
1	(2)		1
2			A

=

The symbol ‘=’ indicates that the chain is a repetition of an earlier one. There is no need to process this further and we can effectively regard it as ‘dead’.

N = 3

1AAA1 → 2AA1
1BB1 → 3B1
1BABA1 → 3ABA1
2BB2
2BABA2 → 2BAB1 → 2BA3
3AAA3
3BB3 → 1B3
3BABA3 → 1ABA3 → 2BA3

	A	B	
1	(2)	(3)	1
2			A
3		1	B

×

=

N = 4

1AAA1 → 2AA1 → 4A1
1BABA1 → 3ABA1
2BB2
2BABA2 → 2BAB1 → 2BA3
3AAA3
4AAA4 → 1AA4 → 2A4
4BB4
4BABA4 → 4BAB2

	A	B	
1	(2)	(3)	1
2	(4)		A
3		1	B
4	1		2A

×

N = 5

1BABA1 → 3ABA1			A	B	
2BB2 → 5B2	×	1	(2)	(3)	1
2BABA2 → 2BAB1 → 2BA3 → 5A3	×	2	(4)	(5)	A
3AAA3 → 3AA5		3		1	B
4BB4		4	1		2A
4BABA4 → 4BAB2 → 4BA5		5	3	2	2B
5AAA5 → 3AA5					

5BB5 → 2B5	×				
5BABA5 → 2ABA5 → 4BA5					

N = 6

1BABA1 → 3ABA1 → 6BA1 → 4A1	×
3AAA3 → 3AA5 → 6A5	×
4BB4 → 5B4	×
4BABA4 → 4BAB2 → 4BA5 → 4B6	×
5AAA5 → 3AA5 → 6A5	×
5BABA5 → 2ABA5 → 4BA5 → 6A5	×
6AAA6 → 5AA6 → 3A6	×
6BB6 → 4B6	×
6BABA6 → 4ABA6 → 1BA6 → 3A6	×

	A	B	
1	(2)	(3)	1
2	(4)	(5)	A
3	(6)	1	B
4	1	6	2A
5	3	2	2B
6	5	4	3A

The first column of the group table is easy, and since $2 = A$ and $3 = B$ the next two columns can be copied from the link table.

Now $4 = 2A$ so we take the '2' column and multiply each entry by A, using the Link Table. So 2, 4, 6, 1, 3, 5

becomes 4, 1, 5, 2, 6, 3. The A column is a permutation of {1, 2, 3, 4, 5, 6} and we are simply using A to permute the ‘2’ column

GROUP TABLE

	1	2	3	4	5	6
1	1	2	3	4		
2	2	4	5	1		
3	3	6	1	5		
4	4	1	6	2		
5	5	3	2	6		
6	6	5	4	3		

Since $5 = 2B$ we permute the ‘2’ column of the group table by the B column of the Link Table, considered as a permutation. This gives us the ‘5’ column. Similarly $6 = 3A$ so we permute the ‘3’ column by A.

GROUP TABLE

	1	2	3	4	5	6
1	1	2	3	4	5	6
2	2	4	5	1	6	3
3	3	6	1	5	4	2
4	4	1	6	2	3	5
5	5	3	2	6	1	4
6	6	5	4	3	2	1

We can always complete the last column by writing down the missing symbol in each row.

Example 15: $G = \langle A, B \mid A^4, B^4, A^2 = B^2, B^{-1}AB = A^{-1} \rangle$. We write the third relator as $AAbb$, and the fourth as $bABA$. This time we present the finished computation, though to help you follow it we enclose in parentheses those links in the link table that came about by creation, rather than the contraction of a chain. Also, since our relators involve the inverse ‘ b ’ we include a column in the link table for it. Remember that every link that goes in the B column will produce an equivalent one in the b column.

CHAIN TABLE	
1AAAAA1 → 2AAA1 → 4AA1 → 7A1	×
1BBBB1 → 3BBB1 → 4BB1 → 8B1	×
1AAbb1 → 2Abb1 → 2Ab3 → 4b3	×
1bABA1 → 1bAB7 → 1bA5 → 8A5	×
2AAAAA2 → 2AAA1	=
2BBBB2 → 5BBB2 → 5BB6 → 7B6	×
2AAbb2 → 4Abb2 → 4Ab5 → 7b5	×
2bABA2 → 2bAB1 → 6AB1 → 6A8	×
3AAAAA3 → 3AAA5 → 6AA5 → 8A5	×
3BBBB3 → 3BBB1	=
3AAbb3 → 3AAb4 → 6Ab4 → 8b4	×
3bABA3 → 1ABA3 → 2BA3 → 5A3	×
4AAAAA4 → 4AAA2 → 4AA1	=
4BBBB4 → 4BBB3 → 4BB1	=
4AAbb4 → 7Abb4 → 1bb4 → 8b4	×
4bABA4 → 3ABA4 → 3AB2 → 6B2	×

5AAAA5→3AAA5	=
5BBBB5→5BBB2	=
5AAbb5→3Abb5→6bb5→6b7	×
5bABA5→2ABA5→4BA5→8A5	×
6AAAA6→6AAA3→6AA5	=
6BBBB6→2BBB6→5BB6→7B6	×
6AAbb6→6AAb2→6AA5	=
6bABA6→6bAB3→6bA1→7A1	×
7AAAA7→1AAA7→2AA7→4A7	×
7BBBB7→6BBB7→2BB7→5B7	×
7AAbb7→1Abb7→2bb7→6b7	×
7bABA7→5ABA7→3BA7→4A7	×
8AAAA8→5AAA8→3AA8→6A8	×
8BBBB8→1BBB8→3BB8→4B8	×
8AAbb8→5Abb8→3bb8→1b8	×
8bABA8→4ABA8→7BA8→6A8	×

LINK TABLE

	A	B	b	
1	(2)	(3)	8	1
2	(4)	(5)	6	A
3	(6)	4	1	B
4	(7)	(8)	3	2A
5	3	7	2	2B
6	8	2	7	3A
7	1	6	5	4A
8	5	1	4	4B

We can obtain the group table from the link table in the usual way. This group is the **quaternion group** of order 8.

GROUP TABLE

	1	2	3	4	5	6	7	8
1	1	2	3	4	5	6	7	8
2	2	4	5	7	8	3	1	6
3	3	6	4	8	2	7	5	1
4	4	7	8	1	6	5	2	3
5	5	3	7	6	4	1	8	2
6	6	8	2	5	1	4	3	7
7	7	1	6	2	3	8	4	5
8	8	5	1	3	7	2	6	4

Example 16: $G = \langle A, B \mid A^3, B^3, (AB)^2 \rangle$

CHAIN TABLE			
1AAA1	\rightarrow 2AA1	\rightarrow 4A1	\times
1BBB1	\rightarrow 3BB1	\rightarrow 7B1	\times
1ABAB1	\rightarrow 2BAB1	\rightarrow 5AB1	\rightarrow 5A7
2AAA2	\rightarrow 2AA1		$=$
2BBB2	\rightarrow 5BB2	\rightarrow 9B2	\times
2ABAB2	\rightarrow 4BAB2	\rightarrow 8AB2	\rightarrow 8A9
3AAA3	\rightarrow 6AA3	\rightarrow 10A3	\times
3BBB3	\rightarrow 3BB1		$=$
4AAA4	\rightarrow 1AA4	\rightarrow 2A4	\times
4BBB4	\rightarrow 4BB6	\rightarrow 8B6	\times
4ABAB4	\rightarrow 1BAB4	\rightarrow 3AB4	$=$

5AAA5→7AA5→11A5	×
5BBB5→5BB2	=
5ABAB5→5ABA2→5AB1	=
6AAA6→6AA3	=
6BBB6→4BB6	=
6ABAB6→6ABA8→10BA8→12A8	×
7AAA7→7AA5	=
7BBB7→1BB7→3B7	×
7ABAB7→7ABA3→7AB10→11B10	×
8AAA8→9AA8→12A8	×
8BBB8→6BB8→4B8	×
8ABAB8→8ABA4→8AB2→9B2	×
9AAA9→9AA8	=
9BBB9→2BB9→5B9	×
9ABAB9→9ABA5→9AB11→12B11	×
10AAA10→3AA10→6A10	×
10BBB10→10BB11→10B12	×
10ABAB10→3BAB10→7AB10	=
11AAA11→5AA11→7A11	×
11BBB11→10BB11	=
11ABAB11→5BAB11→9AB11	=
12AAA12→8AA12→9A12	=
12BBB12→11BB12→10B12	=
12ABAB12→8BAB12→6AB12→10B12	=

LINK TABLE

A B

1	(2)	(3)	1
2	(4)	(5)	A
3	(6)	(7)	B
4	1	(8)	2A
5	7	(9)	2B
6	(10)	4	3A
7	(11)	1	3B
8	9	6	4B
9	(12)	2	5B
10	3	12	6A
11	5	10	7A
12	8	11	9A

GROUP TABLE

	1	2	3	4	5	6	7	8	9	10	11	12
1	1	2	3	4	5	6	7	8	9	10	11	12
2	2	4	5	1	8	7	9	3	6	11	12	10
3	3	6	7	10	4	11	1	12	8	5	2	9
4	4	1	8	2	3	9	6	5	7	12	10	11
5	5	7	9	11	1	12	2	10	3	8	4	6
6	6	10	4	3	12	1	8	7	11	2	9	5
7	7	11	1	5	10	2	3	9	12	4	6	8
8	8	9	6	12	2	10	4	11	5	3	1	7
9	9	12	2	8	11	4	5	6	10	1	7	3
10	10	3	12	6	7	8	11	4	1	9	5	2
11	11	5	10	7	9	3	12	1	2	6	8	4
12	12	8	11	9	6	5	10	2	4	7	3	1

This group is A_4 .

§ 5.5. Why Does It Work?

At this stage it's difficult to give a rigorous proof that the Todd-Coxeter algorithm, when it terminates normally, gives the correct result because we have not yet properly defined what a group presentation really is. But here is an informal proof which will suffice for now.

Suppose we carry out the Todd-Coxeter algorithm on a presentation for a group G and it terminates normally. We end up with a finite number of codes, N , and a link table which expresses each of the generators as a permutation on $\{1, 2, \dots, N\}$. These permutations generate a permutation group which will satisfy all the relations given in the presentation. Provided that different codes correspond to different elements this permutation group will be isomorphic to the group presented by the presentation.

Now identifications can occur during the course of the algorithm, forced by contradictions that arise from interactions between the relations. But if at the end we have two different codes for the same group element we would have an unforced identification, that is, one that is not a consequence of the relations.

But the group described by the presentation is one in which the *only* relations that hold are those that are consequences of the ones given in the presentation. An unforced identification would correspond to a relation that is not a consequence of these. It therefore could not arise. Hence the permutation group obtained at the end is indeed isomorphic to the group being presented.

§ 5.6. Compact Todd-Coxeter

There is a lot of rewriting involved as the chains are born, contract and die. The following is a more compact way of setting out the working. Consider the following reworking of Example 13.

Example 13 (compact): $G = \langle A, B \mid A^2, B^2, (AB)^2 \rangle$.

The Chain Table begins as follows.

A	A	B	B	A	B	A	B
1		1		1			1

Since we don't know how big the group is we don't know how many rows there will be. But the number of columns will remain the same. It is determined by the presentation.

The Link Table begins as follows.

	A	B
1		

Again the number of columns is fixed but we don't know until we are finished how many rows there will be.

The next available code is 2 and we decide to assign it to 1A. So we write (2) in the 1A position of the Link Table. This represents the link 1A2. The reason for the parentheses is to record the fact that this was a choice. So the Link Table now becomes as follows.

	A	B
1	(2)	

Going back to the Chain Table we write 2 where there is a '1' in a ↓ column and an A above. This corresponds to the use of the link 1A2, where $1A = 2$.

A	A	B	B	A	B	A	B
1	2	1		1	2		1

At the same time we create a second row.

A	A	B	B	A	B	A	B
1	2	1		1	2		1
2		2		2			2

We now have the link 2A1 in the top row. We transfer this to the Link Table, which becomes:

	A	B
1	(2)	
2	1	

This time we don't enclose the '1' in parentheses because we didn't choose that link. It was a consequence of the previous choice.

Now we can use that link to further process the Chain Table.

A	A	B	B	A	B	A	B
1	2	1		1	2		1
2	1	2		2	1		2

No further contractions are possible so we introduce a 3rd row.

A	A	B	B	A	B	A	B
1	2	1		1	2		1
2	1	2		2	1		2
3		3		3			3

Let's choose 1B3. This is entered into the Link Table.

	A	B
1	(2)	(3)
2	1	
3		

We can use this new link 1B3 to process the Chain Table further.

A	A	B	B	A	B	A	B
1	2	1	3	1	2		1
2	1	2		2	1	3	2
3		3		3			3

This now gives a new link 3B1. The Link Table becomes:

	A	B
1	(2)	(3)
2	1	
3		1

Back to the Chain Table.

A	A	B	B	A	B	A	B
1	2	1	3	1	2	3	1
2	1	2		2	1	3	2
3		3	1	3		2	1

We were able to put the 3B1 at the end of the top row and the 2A1 in the bottom row, but that's as far as we can go at this stage as we don't know 2B? ?A3 3A? or ?B2. Looks like we'll have to create a '4'

	A	B
1	(2)	(3)
2	1	(4)
3		1
4		

A	A	B	B	A	B	A	B
1	2	1	3	1	2	4	3
2	1	2	4	2	1	3	4
3	4	3	1	3	4	2	1
4	3	4	2	4	3	1	4

We can now complete the Link Table:

	A	B
1	(2)	(3)
2	1	(4)
3	4	1
4	3	2

The Group Table can be obtained as before.

Example 15 (compact):

$$G = \langle A, B \mid A^4, B^4, A^2 = B^2, B^{-1}AB = A^{-1} \rangle.$$

Rather than writing out the Chain Table and Link table at each stage I'll just write out the final tables when the algorithm terminates.

A	A	A	A	B	B	B	B	A	A	A	b	b	b	b	A	B	A
1	2	4	7	1	3	4	8	1	2	4	3	1	8	5	7	1	
2	4	7	1	2	5	7	6	2	4	7	5	2	6	8	1	2	
3	6	8	5	3	4	8	1	3	6	8	4	3	1	2	5	3	
4	7	1	2	4	8	1	3	4	7	1	8	4	3	6	2	4	
5	3	6	8	5	7	6	2	5	3	6	7	5	2	4	8	5	
6	8	5	3	6	2	5	7	6	8	5	2	6	7	1	3	6	
7	1	2	4	7	6	2	5	7	1	2	6	7	5	3	4	7	
8	5	3	6	8	1	3	4	8	5	3	1	8	4	7	6	8	

	A	B	b		A	B	b	
1	(2)	(3)	8		5	3	7	2
2	(4)	(5)	6		6	8	2	7
3	(6)	4	1		7	1	6	6
4	(7)	(8)	3		8	5	1	4

§ 5.7. Marriage Strategies

Probably all of the algorithms you have met so far in your mathematical career are deterministic. This means that the algorithm tells you precisely what to do at each stage and terminates after a finite number of steps. It can therefore be programmed on a computer so that, after feeding in the input data, we simply need to click “GO” and wait for the output.

By contrast the Todd-Coxeter algorithm is a wild beast. It is non-deterministic. Every so often the algorithm consults you as an external ‘oracle’. “I can’t proceed”, it says, “please choose a marriage partner for the next code”. This you do and then it goes off happily for a while until it has to come back for some more advice.

The following three strategies are useful.

MARRIAGE STRATEGY I

Fill up the first blank cell in the Link Table.

For example if our Link Table, at a given stage, is:

	A	B
1	(2)	(3)
2	3	
3		

and we are about to create the code ‘4’ we would, following this strategy, create the link 2B4 by writing (4) into the table. Our Link Table would now be:

	A	B
1	(2)	(3)
2	3	(4)
3		

MARRIAGE STRATEGY II

Choose a link that shortens one of the shortest chains.

For example if we have the chains 1AAAA3, 2ABA3, 1AABB2, and we are about to create the code ‘5’ we would, using this principle, create the link 2A5, or perhaps 5A3. Either way we would now have a chain of length 2 (either 5BA3 or 2AB5, depending which of the alternatives we chose. Short chains are more likely to give links in the near future and links allow us to shorten other chains, so this seems a sensible strategy.

MARRIAGE STRATEGY III

Choose a link that will shorten the most chains.

For example if we have the chains 1AAAA3, 2ABA3, 1AABB2 and we are about to create the code ‘5’ we would, using this principle, create the link 1A5 to shorten the first and third chains, or perhaps 5A3 to shorten the first and second.

In many cases there will be equally attractive alternatives. We can often choose between them by considering which best satisfies the other strategies. For

example using strategy III above we could have created the link 1A5 or 5A3 but since 5A3 also shortens the shortest chain that is the one we should probably choose. There are many other, less obvious strategies. In fact no finite list of strategies will work in all cases.

§ 5.8. Unsolvability of the Word Problem

Which of the above marriage strategies is the best? If we could decide that then we could then build this into the algorithm and so make it deterministic. Unfortunately it is not possible!

The Todd-Coxeter Algorithm (either the restricted or the full version) will, if it terminates normally, give the correct group table and the full version, involving identifications to resolve contradictions, will only ever terminate normally – if it terminates at all. The intrinsic problem with the Todd-Coxeter Algorithm is that it may never terminate.

Of course, if the presentation is that of an infinite group then quite clearly it can never terminate. We would be forced to keep giving out more and more integer codes.

But what if the group is indeed finite? If the algorithm doesn't terminate there is no way of knowing this. Even if the algorithm continues for an enormous length of time, with integer codes running in to the millions, we cannot be sure that it is destined to go on forever. We may be just about to conclude that we have an infinite group when suddenly all the chains collapse and the algorithm terminates. It might just be a very, very

large group, but not an infinite one. Or we could get contradictions and identifications and it might turn out that all the millions of codes represent the same element and we have the trivial group! We can never tell what's around the corner.

On the other hand, even though we have a finite group, because of the choices that we make, the algorithm may go on indefinitely. Certainly it's very easy to make poor choices that would cause this to happen.

What if the group really is finite and we make the 'right' choices? Then it is true that the full algorithm will terminate, giving the correct group table. But how can we know what are the right choices? Is there is a strategy, either a combination of the three given above, or perhaps some other fiendishly clever one, that would guarantee that the full algorithm would always terminate? The answer is a resounding 'NO'. In fact we can't guarantee that, given a presentation, we can always determine whether a given word will reduce to the identity!

It is not simply that nobody has yet come up with such a deterministic algorithm. It has been *proved* that nobody ever will! Like the Halting Problem in computer science, the Word Problem is **unsolvable**.



THE WORD PROBLEM

Construct an algorithm which, given any presentation for a group, and any word in those generators, will determine whether or not that word represents the identity.

Theorem 1: The Word Problem is Unsolvable (i.e. no such algorithm can ever exist).

‘Proof’: This has been proved but, as the proof is rather long and technical, I omit it. Basically one assumes that such an algorithm exists and then proceeds to obtain a logical contradiction.

What it means is that the Todd-Coxeter algorithm attempts to do something that is intrinsically impossible. Yet it works surprisingly well, for ‘reasonable’ presentations, using ‘sensible’ choice strategies (like the three given above). Needless to say, much research has gone into investigating the effect of different choice strategies for different types of presentation.

Can we avoid identifications by making the ‘right’ choices? This is easily answered. We cannot. For example the presentation $\langle A | A^5, A^7 \rangle$ represents the trivial group $\{1\}$. This is because $A = A^{15-14} = (A^5)^3(A^{-7})^2 = 1$. Yet we have to begin by introducing the integer code ‘2’, which ultimately must be identified with ‘1’.

In many other cases, however, when a contradiction arises, it is possible to avoid identifications by starting again and making better choices.

Example 17: $G = \langle A, B \mid A^7, A^2B^2 \rangle$.

If we adopted strategy I at each step we would find that we get a contradiction after having introduced code 21. If you want to recreate this computation the created links would be:

1A2, 1B3, 2A4, 2B5, 3A6, 3B7, 4A8, 4B9, 5A10, 5B11, 6B12, 7B13, 8A14, 8B15, 9A16, 10B17, 11B18, 12A19, 12B20, 14B21.

If we adopted strategy II at each stage we'd find that we would get a contradiction after having introduced code 20. The created links would be:

1A2, 2A3, 3B4, 3A5, 5B6, 4A7, 7A8, 5A9, 6A10, 10A11, 8A12, 9A13, 12A14, 12B15, 13A16, 1B17, 2B18, 11A19, 14A20.

If we adopted strategy III at each stage we would find that we would get a contradiction after having introduced code 17. The created links would be:

1A2, 2A3, 3A4, 4A5, 5A6, 6A7, 1B8, 1b9, 3b10, 5b11, 7b12, 2b13, 4b14, 9a15, 10a16, 10A17.

But the following combination of all three basic strategies terminates without recourse to identifications.

We begin by using strategy III at each stage.

1AAAAAAA1 → 2AAAAAA1 → 3AAAAA1 → 4AAA1 → 5AA1 → 6AA1 → 7A1	×
1AABB1 → 2ABB1 → 3BB1	

2AAAAAAA2	\rightarrow	3AAAAAA2	\rightarrow	3AAAAA1	=				
2AABB2	\rightarrow	3ABB2	\rightarrow	4BB2	=				
3AAAAAAA3	\rightarrow	3AAAAAA2	\rightarrow	3AAAAA1	=				
3AABB3	\rightarrow	4ABB3	\rightarrow	5BB3	=				
4AAAAAAA4	\rightarrow	4AAAAAA3	\rightarrow	4AAAAA2	=				
	\rightarrow	4AAAA1							
4AABB4	\rightarrow	5ABB4	\rightarrow	6BB4	=				
5AAAAAAA5	\rightarrow	5AAAAAA4	\rightarrow	5AAAAA3	=				
	\rightarrow	5AAA2	\rightarrow	5AA1					
5AABB5	\rightarrow	6ABB5	\rightarrow	7BB5	=				
6AAAAAAA6	\rightarrow	6AAAAAA5	\rightarrow	6AAAAA4	=				
	\rightarrow	6AAA3	\rightarrow	6AA2	\rightarrow	6AA1			
6AABB6	\rightarrow	7ABB6	\rightarrow	1BB6	=				
7AAAAAAA7	\rightarrow	1AAAAAA7	\rightarrow	2AAAAA7	=				
	\rightarrow	3AAA7	\rightarrow	4AA7	\rightarrow	5AA7	\rightarrow	6A7	
7AABB7	\rightarrow	1ABB7	\rightarrow	2BB7					

A	B	
1	(2)	1
2	(3)	1A
3	(4)	2A
4	(5)	3A
5	(6)	4A
6	(7)	5A
7	1	6A

We have lots of lovely short chains of the form xBBy but without a B link we'll have no chance of making links from them. For this reason we now switch

to strategy I and fill up the first blank in the B column by creating the link 1B8.

3BB1
4BB2
5BB3
6BB4
7BB5
1BB6→8B6
2BB7
8AAAAAAA8
8AABB8

×

	A	B	
1	(2)	(8)	1
2	(3)		1A
3	(4)		2A
4	(5)		3A
5	(6)		4A
6	(7)		5A
7	1		6A
8		6	1B

It would be very natural now to concentrate on Bs but it turns out that this is not a good idea. Instead we go back to strategy III. It might seem a bad idea to be again filling up the A column and neglecting the B column, but watch!

3BB1	
4BB2	
5BB3	
6BB4	
7BB5	
2BB7	
8AAAAAAA8 → 9AAAAAA8 → 10AAAAA8 → 11AAAA8 → 12AAA8 → 13AA8	×
8AABB8 → 8AAB1 → 9AB1 → 10B1	=
9AAAAAAA9 → 9AAAAAA8 → 10AAAAA8	=
9AABB9 → 10ABB9 → 11BB9	
10AAAAAAA10 → 10AAAAAA9 → 10AAAAA8	=
10AABB10 → 11ABB10 → 12BB10	=
11AAAAAAA11 → 11AAAAAA10 → 11AAAAA9 → 11AAAA8	=
11AABB11 → 12ABB11 → 13BB11	
12AAAAAAA12 → 12AAAAAA11 → 12AAAAA10 → 12AAAA9 → 12AAA8	=
12AABB12 → 13ABB12	
13AAAAAAA13 → 13AAAAAA12 → 13AAAAA11 → 13AAAA10 → 13AAA9 → 13AA8	=
13AABB13	

	A	B	
1	(2)	(8)	1
2	(3)		1A
3	(4)		2A
4	(5)		3A
5	(6)		4A
6	(7)		5A
7	1		6A
8	(9)	6	1B
9	(10)		8A
10	(11)		9A
11	(12)		10A
12	(13)		11A
13			12A

It looks as though we are still a long way off getting rid of all of our chains. But suddenly everything collapses at the very next step!

3BB1→10B1	×
4BB2→11B2	×
5BB3→12B3	×
6BB4→13B4	×
7BB5→14B5	×
2BB7→9B7	×
13AA8→14A8	×
11BB9→2B9	×
12BB10→3B10	×

13BB11→4B11	×
13ABB12→14BB12→5B12	×
13AABB13→14ABB13→8BB13→6B13	×
14AAAAAAA14→8AAAAAA14→9AAAAA14 →10AAAA14→11AAA14→12AA14→13A14	=
14AABB14→8ABB14→9BB14→7B14	×

	A	B	
1	(2)	(8)	1
2	(3)	9	1A
3	(4)	10	2A
4	(5)	11	3A
5	(6)	12	4A
6	(7)	13	5A
7	1	14	6A
8	(9)	6	1B
9	(10)	7	8A
10	(11)	1	9A
11	(12)	2	10A
12	(13)	3	11A
13	(14)	4	12A
14	8	5	13A

I'll now rework this using the compact notation.
But I'll use the same marriage choices.

A A A A A A A A A A B B

1	2	3	4	5	6	7	1	2	3	10	1
2	3	4	5	6	7	1	2	3	4	11	2
3	4	5	6	7	1	2	3	4	5	12	3
4	5	6	7	1	2	3	4	5	6	13	4
5	6	7	1	2	3	4	5	6	7	14	5
6	7	1	2	3	4	5	6	7	1	8	6
7	1	2	3	4	5	6	7	1	2	9	7
8	9	10	11	12	13	14	8	9	10	1	8
9	10	11	12	13	14	8	9	10	11	2	9
10	11	12	13	14	8	9	10	11	12	3	10
11	12	13	14	8	9	10	11	12	13	4	115
12	13	14	8	9	10	11	12	13	14	5	12
13	14	8	9	10	11	12	13	14	8	6	13
14	8	9	10	11	12	13	14	8	9	9	14

	A	B
1	(2)	(8)
2	(3)	9
3	(4)	10
4	(5)	11
5	(6)	12
6	(7)	13
7	1	14

	A	B
8	(9)	6
9	(10)	7
10	(11)	1
11	(12)	2
12	(13)	3
13	(14)	4
14	8	5

How did we know that this combination of strategies was going to work? Perhaps intuition. But of course nothing beats good old group *theory*. In many

cases we can manipulate the presentation and get our answer much more quickly.

In this case we have $A = A^{8-7} = A^8 = (A^2)^4 = (B^{-2})^4 = B^{-8}$
so $G = \langle B \mid B^{-56}, B^{-14} \rangle$
 $= \langle B \mid B^{14} \rangle$, the cyclic group of order 14.

§ 5.9. Direct Products

The **direct product** of two groups G, H is the set of ordered pairs where the first component comes from G and the second from H , that is,

$$\{(g, h) \mid g \in G, h \in H\}.$$

It is denoted by $G \times H$.

Multiplication in $G \times H$ is component-wise, that is

$$(g_1, h_1).(g_2, h_2) = (g_1g_2, h_1h_2).$$

The identity is $(1, 1)$ and $(g, h)^{-1} = (g^{-1}, h^{-1})$.

Strictly speaking the operation \times is not commutative in that $H \times G$ is usually different to $G \times H$. However they are isomorphic, by matching (g, h) in $G \times H$ with (h, g) in $H \times G$.

If the groups are written additively we call it the **direct sum** and write it as $G \oplus H$. Multiplication is defined by $(g_1, h_1) + (g_2, h_2) = (g_1 + g_2, h_1 + h_2)$. The identity is $(0, 0)$ and the inverse of (g, h) is $(-g, -h)$.

If G, H are finite then the order of $G \times H$ (or $G \oplus H$ if the groups are written additively) is $|G| \cdot |H|$.

Within $G \times H$ there is a subgroup $\{(g, 1) \mid g \in G\}$ that is isomorphic to G and another subgroup

$$\{(1, h) \mid h \in H\}$$

that is isomorphic to H . Moreover any element of the first subgroup commutes with any element of the second since $(g, 1)(1, h) = (g, h) = (1, h)(g, 1)$.

Example 18:

Let $G = S_4$, the symmetric group on $\{1, 2, 3, 4\}$ and $H = Z_7^\#$, the set $\{1, 2, 3, 4, 5, 6\}$ under multiplication modulo 7. Then $|G \times H| = 24 \times 6 = 144$.

An example of multiplication is:

$$\begin{aligned} ((12)(34), 5).((123), 4) &= ((12)(34).(123), 5.4) \\ &= ((134), 6) \text{ and} \\ ((1234), 2)^{-1} &= ((1234)^{-1}, 2^{-1}) ((1432), 4). \end{aligned}$$

Example 19: Let $G = \mathbb{Z}$, the group of integers under addition and let $H = \mathbf{R}[x]$, the set of real polynomials under addition. Then $G \times H$ is infinite and

$$(5, x + 1).(3, x - 1) = (15, x^2 - 1).$$

Example 20: Let $G = \langle A, B \mid A^4, B^2, ABAB \rangle$ and $H = \langle A \mid A^5 \rangle$. Then $G \times H$ is isomorphic to $\langle A, B, C \mid A^4, B^2, C^5, ABAB, AC = CA, BC = CB \rangle$.

If we have a presentation in which the generators can be split into two subsets such that those in one subset commute with those in the other, and if there are no other

relations that involve generators from both sets, we can split the group as a direct product.

Example 21: $G = \langle A, B \mid A^4, B^2, AB = BA \rangle$.

In this presentation the relation $AB = BA$ states that A commutes with B . Apart from this there's no relation involving both A and B . So the group splits into a direct product:

$G \cong H \times K$ where $H = \langle A \mid A^4 \rangle$ and $K = \langle B \mid B^2 \rangle$. We don't need to use the Todd-Coxeter algorithm to obtain group tables for these cyclic groups. The group table for H is

	1	2	3	4
1	1	2	3	4
2	2	3	4	1
3	3	4	1	2
4	4	1	2	3

and for K it is

	1	2
1	1	2
2	2	1

The elements of $G = H \times K$ are

$(1, 1), (2, 1), (3, 1), (4, 1), (1, 2), (2, 2), (3, 2), (4, 2)$.

Let's code these as follows.

$1 = (1, 1)$	$5 = (1, 2)$
$2 = (2, 1)$	$6 = (2, 2)$
$3 = (3, 1)$	$7 = (3, 2)$
$4 = (4, 1)$	$8 = (4, 2)$

You need to keep track of which groups these come from. So in the coding $7 = (3, 2)$, you need to remember that $7 \in H \times K$, $3 \in H$ and $2 \in K$ and in the coding $1 = (1, 1)$ the three 1's all represent different identity elements. The first is the identity of $H \times K$, the second is the identity of H and the third is the identity of K .

In calculating the group table for G we proceed as in the following examples:

$$2 \times 7 = (2, 1)(3, 2) = (2 \times 3, 1 \times 2) = (4, 2) = 8$$

$$7 \times 8 = (3, 2)(4, 2) = (3 \times 4, 2 \times 2) = (2, 1) = 2$$

So the group table for G is

	1	2	3	4	5	6	7	8
1	1	2	3	3	5	6	7	8
2	2	3	4	1	6	7	8	5
3	3	4	1	2	7	8	5	6
4	4	1	2	3	8	5	6	7
5	5	6	7	8	1	2	3	4
6	6	7	8	5	2	3	4	1
7	7	8	5	6	3	4	1	2
8	8	5	6	7	4	1	2	3

Notice that the top 4×4 portion of the group table for G is simply the group table for H . This is a consequence of the very orderly way in which we ordered the pairs.

X	Y
Y	X

Notice that the next 4×4 block on the right can be obtained by simply adding 4 to each entry in the first block. In fact the entire table is built out of just two 4×4 blocks, X and Y where X is the group table for H and Y is obtained from X by adding 4 to each entry.

Note too that the pattern of these blocks is that of the group table for K. Noticing these patterns we can make very short work of completing the group table for G. (It's made even easier if we do the whole job in a word processor where we can cut and paste!) Patterns like this always occur whenever we have a direct product.

Example 22:

$$G = \langle A, B, C \mid A^4, B^3, C^2, AB = BA, ACAC, BC = CB \rangle.$$

Here there are two relations that say that B commutes with A and with C. Apart from these there's no relation connecting B with either A or C. So the group splits into a direct product:

$$G = H \times K \text{ where } H = \langle A, C \mid A^4, C^2, ACAC \rangle \text{ and}$$

$$K = \langle B \mid B^3 \rangle.$$

We don't need to use Todd-Coxeter on the second factor.

The group table for $\langle B \mid B^3 \rangle$ is:

	1	2	3
1	1	2	3
2	2	3	1
3	3	1	2

Applying the Todd-Coxeter algorithm to
 $H = \langle A, C \mid A^4, C^2, ACAC \rangle$:

1AAAAA1 → 2AAA1 → 4AA1 → 6A1	×
1CC1 → 3C1	×
1ACAC1 → 2CAC1 → 2CA3 → 5A3	×
2AAAA2 → 2AAA1	=
2CC2 → 5C2	×
2ACAC2 → 4CAC2 → 4CA5 → 7A5	×
3AAAA3 → 3AAA5 → 3AA7 → 8A7	×
3CC3 → 3C1	=
3ACAC3 → 3ACA1 → 3AC6 → 8C6	×
4AAAA4 → 4AAA2 → 4AA1	=
4CC4 → 7C4	×
4ACAC4 → 6CAC4 → 6CA7 → 6C8	×
5AAAA5 → 3AAA5	=
5CC5 → 2C5	=
5ACAC5 → 3CAC5 → 1AC5 → 2C5	=
6AAAA6 → 1AAA6 → 2AA6 → 4A6	=
6CC6 → 8C6	=
6ACAC6 → 1CAC6 → 3AC6	=
7AAAA7 → 5AAA7 → 3AA7 → 8A7	=
7CC7 → 4C7	=
7ACAC7 → 5CAC7 → 2AC7 → 4C7	=
8AAAA8 → 7AAA8 → 5AA8 → 3A8	=
8CC8 → 6C8	=
8ACAC8 → 7CAC8 → 4AC8 → 6C8	=

	A	C	
1	(2)	(3)	1
2	(4)	(5)	A
3	(8)	1	C
4	(6)	(7)	2A
5	3	2	2C
6	1	8	4A
7	5	4	4C
8	7	6	3A

Notice that we departed from the default choice strategy a little in that we didn't define 6 as 3A. We defined 6 as 4A because we had a short chain that could be contracted to a link by defining it in this way. The group table for H is:

	1	2	3	4	5	6	7	8
	1	A	C	2A	2C	4A	4C	3A
1	1	2	3	4	5	6	7	8
2	2	4	5	6	7	1	8	3
3	3	8	1	7	6	5	4	2
4	4	6	7	1	8	2	3	5
5	5	3	2	8	1	7	6	4
6	6	1	8	2	3	4	5	7
7	7	5	4	3	2	8	1	6
8	8	7	6	5	4	3	7	1

The elements of $G = H \times K$ are

$$\begin{aligned}(1, 1), (2, 1), \dots, (8, 1), \\ (1, 2), (2, 2), \dots, (2, 8), \\ (3, 1), (3, 2), \dots, (3, 8).\end{aligned}$$

Let's code these as follows:

$1 = (1, 1)$	$9 = (1, 2)$	$17 = (1, 3)$
$2 = (2, 1)$	$10 = (2, 2)$	$18 = (2, 3)$
$3 = (3, 1)$	$11 = (3, 2)$	$19 = (3, 3)$
$4 = (4, 1)$	$12 = (4, 2)$	$20 = (4, 3)$
$5 = (5, 1)$	$13 = (5, 2)$	$21 = (5, 3)$
$6 = (6, 1)$	$14 = (6, 2)$	$22 = (6, 3)$
$7 = (7, 1)$	$15 = (7, 2)$	$23 = (7, 3)$
$8 = (8, 1)$	$16 = (8, 2)$	$24 = (8, 3)$

When we calculate the group table for G we find that it can be built up as a 24×24 table from three 8×8 blocks:

X	Y	Z
Y	Z	X
Z	X	Y

The block X is the group table for H, the block Y is obtained from this by adding 8 to each of the entries in X, and Z is obtained from X by adding 16 to each entry. The pattern for these blocks follows that for the group table for K.

EXERCISES FOR CHAPTER 5

EXERCISE 1: Find the group table of the groups (A), (B) and (C). Use the Restricted Todd-Coxeter using the Compact Notation.

(A): $\langle A, B | A^2, B^3, (BA)^3 \rangle$.

(B): $\langle A, B | A^5, A^3B^2 \rangle$.

(C): $\langle A, B, C | A^2, B^2, C^3, CBCB, AB = BA, AC = CA \rangle$.

EXERCISE 2: Find the group table of:

$\langle A, B | A^4, B^2, BABA^2 \rangle$.

SOLUTIONS FOR CHAPTER 5

EXERCISE 1:

(A): $\langle A, B | A^2, B^3, (BA)^3 \rangle$.

A A B B B B A B A B A

1	2	1	3	6	1	3	5	9	8	2	1
2	1	2	4	8	2	4	7	10	6	1	2
3	5	3	6	1	3	6	10	12	11	5	3
4	7	4	8	2	4	8	9	11	12	7	4
5	3	5	9	11	5	9	8	2	1	3	5
6	10	6	1	3	6	1	2	4	7	10	6
7	4	7	10	12	7	10	6	1	2	4	7
8	9	8	2	4	8	2	1	3	5	9	8
9	8	9	11	5	9	11	12	7	4	8	9
10	6	10	12	7	10	12	11	5	3	6	10
11		11	5	9	11	5	3	6	10	12	11
12	11	12	7	10	12	7	4	8	9	11	12

	A	B
1	(2)	(3)
2	1	(4)
3	(5)	(6)
4	(7)	(8)
5	3	(9)
6	(10)	1

	A	B
7	4	10
8	9	2
9	8	(11)
10	6	(12)
11	12	5
12	11	7

The group table is:

	A	B	2B	3A	3B	4A	4B	5B	6A	9B	10B	
1	1	2	3	4	5	6	7	8	9	10	11	12
2	2	1	4	3	7	8	5	6	10	9	12	11
3	3	5	6	9	10	1	8	11	12	2	7	4
4	4	7	8	10	9	2	6	12	11	1	5	3
5	5	3	9	6	8	11	10	1	2	12	4	7
6	6	10	1	12	2	3	11	7	4	5	8	9
7	7	4	10	8	6	12	9	2	1	11	3	5
8	8	9	2	11	1	4	12	5	3	7	6	10
9	9	8	11	2	12	5	1	4	7	3	10	6
10	10	6	12	1	11	7	2	3	5	4	9	8
11	11	12	5	7	3	9	4	10	6	8	1	2
12	12	11	7	5	4	10	3	9	8	6	2	1

(B): $\langle A, B | A^5, A^3B^2 \rangle$.

	A	A	A	A	A	A	A	A	B	B
1	2	3	4	5	1	2	3	4	9	1
2	3	4	5	1	2	3	4	5	10	2
3	4	5	1	2	3	4	5	1	6	3
4	5	1	2	3	4	5	1	2	7	4
5	1	2	3	4	5	1	2	3	8	5
6	7	8	9	10	6	7	8	9	1	6
7	8	9	10	6	7	8	9	10	2	7
8	9	10	6	7	8	9	10	6	3	8
9	10	6	7	8	9	10	6	7	4	9
10	6	7	8	9	10	6	7	8	5	10

	A	B
1	(2)	(6)
2	(3)	7
3	(4)	8
4	(5)	9
5	1	(10)
6	(7)	3
7	(8)	4
8	(9)	5
9	(10)	1
10	6	2

The group table is:

	A	2A	3A	4A	B	6A	7A	8A	9A
1	2	3	4	5	6	7	8	9	10
1	1	2	3	4	5	6	7	8	9
2	2	3	4	5	1	7	8	9	10
3	3	4	5	1	2	8	9	10	6
4	4	5	1	2	3	9	10	6	7
5	5	1	2	3	4	10	6	7	8
6	6	7	8	9	10	3	4	5	1
7	7	8	9	10	6	4	5	1	2
8	8	9	10	6	7	5	1	2	3
9	9	10	6	7	8	1	2	3	4
10	10	6	7	8	9	2	3	4	5

Note that we had to be canny about choosing the marriage partners. If your computation went beyond 10 it would either eventually reach a contradiction or go on forever. Choice of marriage partners is *so* important!

(C): $\langle A, B, C \mid A^2, B^2, C^3, CBCB, AB = BA, AC = CA \rangle$.
 $\cong \langle A \mid A^2 \rangle \times \langle B, C \mid B^2, C^3, CBCB \rangle$.

B	B	C	C	C	C	B	C	B
1	2	1	3	6	1	3	5	1
2	1	2	4	5	2	4	6	2
3	5	3	6	1	3	6	4	3
4	6	4	5	2	4	5	3	4
5	3	5	2	4	5	2	1	5
6	4	6	1	3	6	1	2	6

	B	C
1	(2)	(3)
2	1	(4)
3	(5)	(6)
4	6	5
5	3	2
6	4	1

The group table for this group is:

	B	C	2C	3B	3C
1	1	2	3	4	5
2	2	1	4	3	6
3	3	5	6	2	4
4	4	6	5	1	3
5	5	3	2	6	1
6	6	4	1	5	2

The group table for the cyclic group is:

	1	A
1	1	A
A	A	1

We code the ordered pairs as follows:

11	12	13	14	15	16	A1	A2	A3	A4	A5	A6
1	2	3	4	5	6	7	8	9	10	11	12

The group table for the direct product has the form:

		1-6	7-12
1-6	T	T + 6	
7-12	T + 6	T	

Where T is the group table for the second group and $T + 6$ is that same table with 6 added to every entry. In full the group table for the entire group is:

	1	2	3	4	5	6	7	8	9	10	11	12
1	1	2	3	4	5	6	7	8	9	10	11	12
2	2	1	4	3	6	5	8	7	10	9	12	11
3	3	5	6	2	4	1	9	11	12	8	10	7
4	4	6	5	1	3	2	10	12	11	7	9	8
5	5	3	2	6	1	4	11	9	8	12	7	10
6	6	4	1	5	2	3	12	10	7	11	8	9
7	7	8	9	10	11	12	1	2	3	4	5	6
8	8	7	10	9	12	11	2	1	4	3	6	5
9	9	11	12	8	10	7	3	5	6	2	4	1
10	10	12	11	7	9	8	4	6	5	1	3	2
11	11	9	8	12	7	10	5	3	2	6	1	4
12	12	10	7	11	8	9	6	4	1	5	2	3

EXERCISE 4: $\langle A, B | A^4 = 1, B^2 = 1, BABAA = 1 \rangle$.

If you were to try the Restricted Todd-Coxeter Algorithm, then whatever you chose for marriage partners you would get contradictions and you would need to abort. I haven't shown you the full version and, if you had used it, making the 'right' choices, you would be able to work out the group table. At certain stages, when you got contradictions you would have to identify two codes as representing the same group element. A lot of work.

But, using a little group theory, you can easily identify this group.

The relations $A^4 = 1$ and $B^2 = 1$ mean that we can write $BABAA = 1$ as $B^{-1}AB = A^2$.

Then $B^{-1}A^2B = A^4 = 1$ and so $A^2 = B^{-1}B = 1$. But then $BAB = 1$. Since $B = B^{-1}$ this becomes $B^{-1}AB = 1$ which gives $A = 1$.

So the group is simply $\langle B \mid B^2 \rangle$, the cyclic group of order 2, and you don't need me to write out its group table.

Sometimes using the Todd-Coxeter Algorithm is like using a sledgehammer to crack a nut. This example highlights the fact that the Todd-Coxeter Algorithm performs badly with an unnecessarily complicated presentation.