AN EXTENSION OF COSET ENUMERATION

by

HARVEY A. CAMPBELL

A thesis submitted to the Faculty of Graduate Studies and Research, in partial fulfilment of the requirements for the degree of Master of Science.

Department of Mathematics
McGill University
Montreal

November, 1971

ABSTRACT

Author: Harvey A. Campbell

Title: An Extension of Coset Enumeration

Department: Mathematics

Degree: Master of Science

The purpose of this thesis is to present the exposition of a new algorithm. If we are given a group and a subgroup of finite index the algorithm determines the index. This part of the algorithm derives from the Todd-Coxeter algorithm. The second function is to give a presentation for the subgroup in terms of those original generators of the subgroup. This part of the algorithm is similar in scope to the classical Reidemeister-Schreier technique. The thesis first gives a detailed description of the new algorithm with a worked example. Then a proof is given that the algorithm does give a presentation for the subgroup with the proof based on a proof used for the Reidemeister-Schreier algorithm. An actual FORTRAN program is included which executes the algorithm. The last chapter is concerned with various examples.

The author wishes to express deep gratitude to a scholar,

Professor W. O. J. Moser

whose ideas and encouragement made this thesis possible.

TABLE OF CONTENTS

	Page
Introduction	1
0: Preliminary Details	2
ONE: Extended Coset Enumeration	5
TWO: Reidemeister-Schreier Algorithm	34
THREE: A Simple Proof	41
FOUR: A Computer Program	48
FIVE: Some Examples	57
Bibliography	76

INTRODUCTION

The purpose of this thesis is to present the exposition of an algorithm. The algorithm performs two functions. If we are given a group and a subgroup of finite index, the algorithm determines the index. The second function is to give a presentation for the subgroup in terms of the original generators of the subgroup. This is the first time the complete algorithm has been described.

This first function is the Todd-Coxeter algorithm (Coxeter & Moser, 1964). The second function is the Reidemeister-Schreier rewriting process (Magnus, Karrass, & Solitar, 1966, pp.86-98). The algorithm we describe combines these two operations into one.

The first chapter of the thesis, called chapter 0, gives a few preliminary details. In chapter one we describe the new algorithm in detail, with a worked example. In the second chapter we outline the Reidemeister-Schreier process with the same example worked out. In the next chapter we prove the algorithm does actually give a presentation of the subgroup. Then we discuss a machine version of the algorithm giving the actual FORTRAN program. The last chapter contains some additional results concerning various presentations. These results show how the algorithm may be used.

CHAPTER 0

Let G be a group and let

$S = \{ x, y, z, \ldots \}$

be an arbitrary set of symbols and

$S^{-1} = \{ x^{-1}, y^{-1}, z^{-1}, \ldots \}$

an associated set of symbols. A word W is a finite sequence of symbols in $S \cup S^{-1}$, and the length L(W), of W is the number of symbols from $S \cup S^{-1}$ in W. We define the empty word as a word of length 0 denoted by 1 or e.

Now let α be a mapping from S to G

α: $x \to g$

α: $y \to h$

.

.

.

We say that under α, x defines g, y defines h, etc... If W is a word in S and

$W = f_1 f_2 \ldots f_n$

and f_i defines g_i for i = 1,2, ... n we say W defines $g_1 g_2 \ldots g_n$. If α is such that every element of G is defined then we say S is a set of defining symbols for G. Suppose W is a word in S such that under α, W defines the identity element of G; then W is called a relator of G. The equation

$R(x, y, z, \ldots) = S(x, y, z, \ldots)$

where the equality means as an image of α is called a relation, if the word RS^{-1} is a relator. An equation of the form

$$x = x$$

is called a trivial relation and the word xx^{-1} is called a trivial relator.

Suppose P, Q, R, ... are relators in G. We say that the word W is derivable from P, Q, R, ... if the following operations applied a finite number of times change W into the empty word:

1. Insertion of one of the words P, P^{-1}, Q, Q^{-1}, ... or one of the trivial relators.
2. Deletion of one of the words P, P^{-1}, Q, Q^{-1}, ... or one of the trivial relators.

Let $\Phi = \{ P, Q, R, \ldots \}$ be a set of relators such that every relator is derivable from Φ, then Φ is said to be a set of defining relators for G. So if we have the sets S, Φ, we write

$$G = \langle\, S \,;\, \Phi \,\rangle$$

and call this a presentation of G. We say the presentation is finitely generated (finitely related) if the number of generators (defining relations) is finite. If it is both, then we say the presentation is finite.

For example, any cyclic group is finitely presented, since if the order of the group is n then

$$\langle\, a \,;\, a^n \,\rangle$$

is a presentation.

It is true that every group has a presentation (not necessarily finite, see Magnus, Karrass & Solitar, 1966).

The distinction between relator and relation may be relaxed since when we have a relator W we also have the relation

$$W = e$$

and conversely.

CHAPTER ONE

Let G be a discrete group generated by a finite number of generators

$$s_1, s_2, s_3, \dots s_m \qquad m \geq 1$$

and defined by a finite number of relations

$$g_i(s) = e \qquad i = 1, 2, \dots r$$

where the $g_i(s)$ are words in the generators s_j and s_j^{-1}, $j = 1, \dots m$.

Let $\qquad t_k = t_k(s) \qquad k = 1, 2, \dots n$

be n words in the generators s_j and s_j^{-1} and let

$$H = \{ t_1, t_2, \dots t_n \}$$

be the subgroup generated by these words. The problems we wish to solve are :

1. Assuming the index of H in G to be finite we wish to find the index.
2. To give a presentation for H.

Concerning problem 1, E. H. Moore (1897) and many others have systematically enumerated the cosets of H in G. Todd and Coxeter (1936) converted this method into a mechanical technique. It is precisely this technique upon which the more sophisticated algorithm that we describe is based.

Problem two requires us to find relations in the symbols t_k such that every relation in H is a consequence of these relations. We will consider the classical solution to problem two in chapter two.

We will use the positive integers *1, 2, 3, ...* to denote the cosets of H in G, and always agree to let the coset H be denoted by *1*. We shall also be choosing coset representatives (one representative for each coset) and agree to denote the representative chosen for coset i by the symbol $\bar{i}$. We always choose $\bar{1}$ to be the identity. The representatives $\bar{2}$, $\bar{3}$, $\bar{4}$, ... are, of course, words in the generators s_j and their inverses s_j^{-1}.

As the computation of the algorithm proceeds we build several tables, containing various sorts of information obtained to that point. The first table contains all the information of the type

$$is_j = k, \qquad ks_j^{-1} = i$$

i.e. how the cosets are permuted among themselves on right multiplication by the generators of G and their inverses. This information is completely contained in a table displayed as follows:

	s_1	s_2	. . .	s_m	s_1^{-1}	s_2^{-1}	. . .	s_m^{-1}
1								
2								
3								
.								
.								
.								

TABLE I

The entries are integers (cosets) and occur at the intersection of the rows indexed by *1*, *2*, *3*, ... and columns headed by s_1, s_2, s_3, ... s_1^{-1}, s_2^{-1}, s_3^{-1}, ... The information

$$is_j = k$$

is recorded by placing the integer k at the intersection of the i th row and s_j th column. The information

$$ks_j^{-1} = i$$

is recorded by placing the integer i in the kth row under the column headed by s_j^{-1}. When the algorithm is complete, TABLE I will have $d = [G:H]$ rows and there will be no empty spaces in the table.

The second table which we will refer to as TABLE II will contain information of the type

$$\bar{i}s_j = W\bar{k}, \qquad \bar{k}s_j^{-1} = W^{-1}\bar{i}$$

where W is an element of H. That is, when we multiply a coset representative on the right by a generator or its inverse we get a word in the subgroup times another coset representative. Of course, that word may only be the identity. (In the classical method, described further, W is expressed as a word in the Schreier generators of H and their inverses; in our algorithm the words W are expressed as words in the generators t_i and their inverses.) The information is recorded in TABLE II as was done in TABLE I.

Clearly the information contained at any stage in TABLE I is contained in TABLE II but not conversely for,

$$\bar{i}s_j = W\bar{k}$$

implies immediately

$$is_j = k$$

but from the latter the former cannot be deduced immediately. TABLE II is shown on the following page.

	s_1	s_2	. . .	s_m	s_1^{-1}	s_2^{-1}	. . .	s_m^{-1}
1								
2								
.								
.								
.								

TABLE II

Now let W be a word in G or H. Then W is said to be written in expanded form (Coxeter & Moser 1964 p.13) if the exponent of any symbol in W is ±1. For example

$$W = s_1^3 s_5^2 s_7^{-2}$$

written in expanded form is

$$W = s_1 s_1 s_1 s_5 s_5 s_7^{-1} s_7^{-1}$$

The third table consists of n subtables, one for each subgroup generator $t_k = t_k(s)$. If

$$t_k(s) = s_i^a s_j^b \; \ldots \; s_p^f$$

then the table would appear as follows

$$\begin{array}{ccccccccccccccc} & t_k & = & s_i^\circ & \ldots & s_i^\circ & s_j^\circ & \ldots & s_j^\circ & \ldots & s_p^\circ & \ldots & s_p^\circ & & \\ \mathit{1} & | & \mathit{1} & \mathit{1}\,| & & | & | & & | & & | & & | & \mathit{1} & \end{array} \qquad \circ = \pm 1$$

TABLE III

The vertical bars separate the symbols. Now we know that t_k is in the subgroup hence

$$\mathit{1}\, t_k = \mathit{1}$$

and this information has already been entered above. We fill into TABLE III as much information as possible from TABLE I. For example it may be that s_i is already a subgroup generator in which case the above table would look like;

$$\begin{array}{ccccccccccccccc} & t_k & = & s_i^\circ & s_i^\circ & \ldots & s_i^\circ & s_j^\circ & \ldots & s_j^\circ & \ldots & s_p^\circ & \ldots & s_p^\circ & \\ 1| & 1 & & 1| & 1| & 1\ldots 1| & 1| & & & | & & | & & | & 1 \end{array}$$

The fourth and last table consists of r subtables, one for each relator $g_i(s)$. Each relator is written in expanded form with a number of rows below it. For example corresponding to the relator

$$g_i = s_p^d s_q^e \ldots s_t^1$$

we set up the table

$$\begin{array}{l} s_p^\circ\ s_p^\circ \ldots s_p^\circ\ s_q^\circ\ s_q^\circ \ldots s_q^\circ \ldots s_t^\circ \ldots s_t^\circ \qquad \circ = \pm 1 \\ 1 \qquad\qquad\qquad\qquad\qquad\qquad\qquad\quad 1 \\ 2 \qquad\qquad\qquad\qquad\qquad\qquad\qquad\quad 2 \\ \vdots \qquad\qquad\qquad\qquad\qquad\qquad\qquad\quad \vdots \end{array}$$

TABLE IV

Now

$$1g_i = 1$$

so that is why a *1* appears at both ends. In fact for any coset j and any relator g we have

$$jg = j.$$

We fill in as much information as possible from TABLE I.

To see better what these tables look like at the preliminary stage consider the group

$$G = \langle\ A,X;\ X^2AX = A^2\ \rangle$$

and the subgroup

$$H = \{\ t_1 = X,\ t_2 = AXXXXA^{-1},\ t_3 = A^{-1}XXXXA\ \}.$$

We immediately have

$$1 = H \qquad \bar{1} = e$$

$$1X = 1 \qquad \bar{1}X = t_1\bar{1}$$
$$1X^{-1} = 1 \qquad \bar{1}X^{-1} = t_1^{-1}\bar{1}$$

The tables are set up as follows;

TABLE I

	X	X^{-1}	A	A^{-1}
1	1	1		

TABLE II

	X	X^{-1}	A	A^{-1}
$\bar{1}$	$t_1\bar{1}$	$t_1^{-1}\bar{1}$		

TABLE III

$t_1 = X$

$1 \mid 1 \quad 1 \mid 1$

$t_2 = A\ X\ X\ X\ X\ A^{-1}$

$1 \mid 1 \quad 1 \mid \qquad \mid 1$

$t_3 = A^{-1}\ X\ X\ X\ X\ A$

$1 \mid 1 \quad 1 \mid \qquad \mid 1$

TABLE IV

$X\ X\ A\ X\ A^{-1}\ A^{-1}$

$1 \mid 1 \mid 1 \mid \qquad \mid 1$

We leave this example for a moment and continue with our discussion of the general algorithm. The first step in the algorithm is to define new information. This is a basic step at any stage. If all known information has been processed and there are still blank spaces in TABLE I then new cosets are defined. Thus we define coset 2 by setting

$$2 = 1s_j^{\circ} \qquad \circ = \pm 1$$

where s_j is a group generator and the product $1s_j^{\circ}$ is not known. It then follows that

$$1 = 2s_j^{-\circ}.$$

At the same time we choose $\bar{2}$, the representative of coset 2, to be

$$\bar{2} = \bar{1}s_j^{\circ}$$

and as above it follows

$$\bar{1} = \bar{2}s_j^{-\circ}.$$

However we know

$$\bar{1} = e$$

$$\text{so } \bar{2} = s_j^{\circ}$$

We put this information in tables I and II. We also put it in tables III and IV wherever possible. We now create new rows in TABLE IV by entering the new coset in every essentially different place (of course omitting the places where it has already occurred.) By " essentially different " we mean the following:

A relator has a base length (Trotter, 1966,p.13), that is to say a block of symbols of minimal length which is repeated one or more times, to make up the relator. For example in the relator a^n the base length is one and in the relator $(ab)^n$ the base length is two. When we say a coset appears in every essentially different place we mean the coset appears to the left of every symbol and to the right of every symbol in the base. Thus if a relator has a base length of k, each coset must appear k or k+1 times in each subtable in TABLE IV (a coset will appear k+1 times if it commences a row, otherwise k times) We return to our example:

We define

$$2 = 1A$$

$$\bar{2} = \bar{1}A$$

and enter this information in all four tables. The tables appear on the next page.

TABLE I

	X	X^{-1}	A	A^{-1}
1	*1*	*1*	*2*	
2				*1*

TABLE II

	X	X^{-1}	A	A^{-1}
$\bar{1}$	$t_1\bar{1}$	$t_1^{-1}\bar{1}$	$\bar{2}$	
$\bar{2}$				$\bar{1}$

TABLE III

$t_1 = X$

1 | *1*   *1* | *1*

$t_2 = A \ X \ X \ X \ X \ A^{-1}$

1 | *1*   *1* | *2*   . . .   *2* | *1*

$t_3 = A^{-1} \ X \ X \ X \ X \ A$

1 | *1*   *1*   . . .   *1*

TABLE IV

	X	X	A	X	A^{-1}	A^{-1}
1	*1*	*1*	*2*		*2*	*1*
			1	*1*		
				2	*1*	
2						*2*
	2					
		2				

We now discuss what occurs in tables III and IV. There are two possibilities for a particular row. The first is that there are blank spaces. In tables III and IV above every row has blank spaces. In this case nothing can be done so it is left.

The second possibility occurs when the row becomes complete i.e. no blank spaces left. Suppose in our example we now define

$$3 = 2X.$$

We reproduce row *1* from TABLE IV above after entering the information

	X	X	A	X	A^{-1}	A^{-1}
1	*1*	*1*	*2*	*3*	*2*	*1*.

i.e. the row closes! We can now conclude

$$2 = 3A^{-1}$$

$$3 = 2A.$$

We have gained new information. This is called row closure. We put this information in TABLE I and wherever possible in the other rows in tables III and IV. The same procedure is used if row closure occurs in TABLE III.

When new cosets are defined there are no rules to say what the definition should be. However since row closure gives additional information it is usually better to define cosets in such a manner that closure occurs as often as possible. TABLE III is closed first and then the rows in TABLE IV starting with row *1*.

When row closure occurs, additional information of the type contained in TABLE II is also obtained. There are two separate cases. If row closure occurs in TABLE IV we procede as follows:

We had

$$\begin{array}{ccccccc} & X & X & A & X & A^{-1} & A^{-1} \\ 1 & 1 & 1 & 2 & 3 & 2 & 1 \end{array}$$

We solve for the generator where closure occurred, in this case A^{-1}

$$A^{-1} = X^{-1} A^{-1} X^{-1} X^{-1} A$$

Now we multiply both sides of the equation on the left by the coset occurring to the left of the solved for generator

$$\bar{3}A^{-1} = \bar{3}\, X^{-1} A^{-1} X^{-1} X^{-1} A$$

We go to TABLE II and look up the information required and we find

$$\bar{3}\, X^{-1} = \bar{2}$$

So, continuing to multiply in this manner

$$\begin{aligned} \bar{3}\, A^{-1} &= \bar{2}\, A^{-1} X^{-1} X^{-1} A \\ &= \bar{1}\, X^{-1} X^{-1} A \\ &= t_1^{-1}\, \bar{1}\, X^{-1} A \\ &= t_1^{-1} t_1^{-1}\, \bar{1}\, A \\ &= t_1^{-2}\, \bar{2} \end{aligned}$$

Thus we get the new information

$$\bar{3}\, A^{-1} = t_1^{-2}\, \bar{2}$$

$$\bar{2}\, A = t_1^{2}\, \bar{3}$$

This information is put in TABLE II. Without reference to a particular example the following points should be noted:

1. All products of the form $\bar{i}\cdot s$ are known because the row closed, i.e. the rewriting can always be carried out.
2. If the final information is

   $$\bar{i}\ s = W\ \bar{j}$$

   then W is a word in the subgroup.
3. This procedure is called rewriting a row i. e. rewriting in terms of subgroup symbols and a coset representative.

If row closure occurs in TABLE III the same procedure is followed with a slight complication. The complication is due to the fact that we are dealing with a subgroup generator of the form

$$t_k = t_k(s)$$

When we solve for the generator where closure occurred, we must remember to include t_k on the left hand side of the equation. To show the exact procedure we will consider the subgroup generator t_2 in our example

	t_2	=	A		X		X		X		X		A^{-1}	
1		1		1		2		3				2		1

We have entered all the information that is possible at this moment. We now define

$$3X = 4 \qquad \bar{3}X = \bar{4}$$

$$4X = 5 \qquad \bar{4}X = \bar{5}$$

From the definitions we get immediately

$$4X^{-1} = 3 \qquad \bar{4}X^{-1} = \bar{3}$$

$$5X^{-1} = 4 \qquad \bar{5}X^{-1} = \bar{4}$$

We put this information in tables I and II and proceed to fill in as much as possible in TABLE III. We consider the generator t_2

$$\begin{array}{ccccccccc} t_2 & = & A & X & X & X & X & A^{-1} & \\ 1 \quad 1 & & 1 & 2 & 3 & 4 & 5 & 2 & 1 \end{array}$$

Row closure occurs in TABLE III. We see immediately

$$5X = 2$$

$$2X^{-1} = 5$$

However as was in the case in TABLE IV there is additional information contained in row closure in TABLE III. We solve for the group generator where closure occurred, in this case, X

$$X = X^{-1}\ X^{-1}\ X^{-1}\ A^{-1}\ \underline{t_2}\ A$$

Note the difference between row closure here and in TABLE IV. Now we proceed as before and multiply both sides of the equation on the left by the appropriate coset representative.

$$\bar{5}X = \bar{5}\ X^{-1}\ X^{-1}\ X^{-1}\ A^{-1}\ t_2\ A$$

As before, we go to TABLE II and get the information necessary to carry out the multiplication on the right hand side.

$$\begin{aligned} \bar{5}X &= \bar{5}\ X^{-1}\ X^{-1}\ X^{-1}\ A^{-1}\ t_2\ A \\ &= \bar{4}\ X^{-1}\ X^{-1}\ A^{-1}\ t_2\ A \\ &= \bar{3}\ X^{-1}\ A^{-1}\ t_2\ A \\ &= \bar{2}\ A^{-1}\ t_2\ A \\ &= \bar{1}\ t_2\ A \\ &= t_2\ \bar{1}\ A \end{aligned}$$

$\bar{5}X = t_2\,\bar{2}$

This information is put in TABLE II. We have now dealt with row closure in tables III and IV.

The example we are dealing with now looks like:

TABLE I

	X	X^{-1}	A	A^{-1}
1	1	1	2	
2	3	5	3	1
3	4	2		2
4	5	3		
5	2	4		

TABLE II

	X	X^{-1}	A	A^{-1}
$\bar{1}$	$t_1\bar{1}$	$t_1^{-1}\bar{1}$	$\bar{2}$	
$\bar{2}$	$\bar{3}$	$t_2^{-1}\bar{5}$	$t_1^2\bar{3}$	$\bar{1}$
$\bar{3}$	$\bar{4}$	$\bar{2}$		$t_1^{-2}\bar{2}$
$\bar{4}$	$\bar{5}$	$\bar{3}$		
$\bar{5}$	$t_2\bar{2}$	$\bar{4}$		

TABLE III

t_1 = X

1 | 1   1 | 1

t_2 = A X X X X A^{-1}

1 | 1   1 | 2 | 3 | 4 | 5 | 2 | 1

t_3 = A^{-1} X X X X A

1 | 1   1 | | 1

TABLE IV

	X	X	A	X	A^{-1}	A^{-1}
1	1	1	2	3	2	1
			1	1		
			5	2	1	
2	3	4			3	2
5	2	3				5
4	5	2	3	4		4
3	4	5				3
			4	5		
					5	
					4	

There is one further situation to be dealt with. Suppose a row closes and we get information of the type

$i s = j.$

We proceed to put j in the ith row and under the s in TABLE I but find the space is not empty! Thus we already have an equation

$$i s = k$$

If $j \neq k$ then we have a redundancy (Mendelsohn,1964, p.509) or a coincidence (Todd & Coxeter, 1936). By these words we mean that two numbers represent the same coset. Thus both j and k stand for the same coset.

The situation is dealt with as follows. We first rewrite the row where closure occurred to get the information

$$\bar{i}\ s = W\ \bar{j}$$

We also look in TABLE II to get the information

$$\bar{i}\ s = V\ \bar{k}$$

Combining the information in an algebraic manner i.e. by equating we find

$$j = k$$

$$W\ \bar{j} = V\ \bar{k}$$

What does this say? Well, if we look in TABLE I it means the row numbered j and the row numbered k are the same. Thus we can compare the two rows, column by column, equating entries. A similar situation occurs in TABLE II. We assume that in numerical value

$$j < k.$$

We have the equation

$$W\ \bar{j} = V\ \bar{k}$$

or

$$\bar{j} = W^{-1} V\ \bar{k}$$

We must now multiply every entry in the $\bar{k}$th row in TABLE II by $W^{-1}V$

and then equate rows $\bar{j}$ and $\bar{k}$.

There are four cases that can occur when comparing the two rows in tables I and II. Suppose we consider a column headed by a group generator s_1. Let us denote by (s_1, j) the entry in the jth row and under s_1 in TABLE I. Similarly let $(s_1, \bar{j})$ denote the entry in the $\bar{j}$th row and under s_1 in TABLE II. The four cases are:

1. (s_1, j) and (s_1, k) are both empty.
2. (s_1, j) is empty and (s_1, k) is not.
3. (s_1, j) is not empty and (s_1, k) is empty.
4. Both (s_1, j) and (s_1, k) are nonempty.

Of course, if (s_1, j) is empty then so is $(s_1, \bar{j})$. Thus the above cases hold in TABLE II as well. They are dealt with as follows:

1. This case is trivial. We also have $(s_1, \bar{j})$ and $(s_1, \bar{k})$ are empty. There is no new information and nothing is done.

2. In this case we have picked up new information. We put the entry (s_1, k) in the blank space (s_1, j). In TABLE II we put the entry $(s_1, \bar{k})$ in the blank space $(s_1, \bar{j})$. It should be remembered we have multiplied the entry $(s_1, \bar{k})$ by $W^{-1}V$ before comparing the rows.

3. There is no new information in this case so nothing is done.

4. This case provides further redundancies for we know that the entry (s_1, j) and the entry (s_1, k) are equal. This information is put in a list to await processing. Thus if we assume the following e.g.

$$(s_1, j) = \ell$$

$$(s_1, k) = m$$

and we also assume e.g.

$$(s_1, \bar{j}) = W_1\bar{\ell}$$
$$(s_1, \bar{k}) = V_1\bar{m}$$

where W_1 and V_1 are words in the subgroup, then we would have the following information in our list

$$\ell = m$$
$$W_1\bar{\ell} = V_1\bar{m}$$

We are now finished with the row k in TABLE I and the row $\bar{k}$ in TABLE II, hence these rows are erased. However there will be occurrences of k in the body of TABLE I and occurrences of $\bar{k}$ in the body of TABLE II.We now go through TABLE I and replace every occurrence of k by j. We also know

$$\bar{j} = W^{-1}V\bar{k}$$

so
$$\bar{k} = V^{-1}W\bar{j}$$

We now go through TABLE II and replace every occurrence of $\bar{k}$ by $V^{-1}W\bar{j}$.

Next we deal with TABLE III. If a row has closed we ignore it even if it contains k. The replacement of k by j would not give us new information. If however k occurs in a row which has not closed then we replace k by j. In TABLE IV we merely erase every row which contains k. We do this because these rows merely duplicate the rows containing j in the same position.

The last step in dealing with redundancies is to look at the list of redundancies waiting to be processed. If there are any further occurrences of k and $\bar{k}$ in the list they are replaced by j and $V^{-1}W\bar{j}$ respectively.

The problem of redundancies would at first glance appear to be fairly complicated. In order to deal properly with redundancies we must remember that we are systematically deleting occurrences of the higher numbered coset in any place it may occur. After the higher number has been deleted, it will leave a gap in the tables. This gap can be filled in two ways. We either renumber the cosets to fill the gap or we define a new coset with the old number.

We are now finished with the first redundancy. We proceed to deal with the next redundancy in the same systematic manner. There is one further situation which may occur in this list. That is, we may have equations of the form

$$j = j$$

$$V\bar{j} = W\bar{j}$$

i.e. a coset equal to itself. This is a very easy matter to deal with. We notice in the equation

$$V\bar{j} = W\bar{j}$$

that we can cancel $\bar{j}$ and we are left with

$$V = W.$$

This is a relation in the subgroup. It will form part of a defining set of relations so it is kept in a list. This is all that is necessary to deal with this situation. After we complete processing of the list of redundancies, we proceed to tables III and IV and fill in as much information as possible.

We consider our example and give some concrete occurrences of redundancies. Let us consider the subgroup generator t_3

$$t_3 = A^{-1}\ X\ X\ X\ X\ A$$

We define

$6 = 1A^{-1}$ $\qquad$ $\bar{6} = \bar{1}A^{-1}$

$7 = 6X$ $\qquad$ $\bar{7} = \bar{6}X$

$8 = 7X$ $\qquad$ $\bar{8} = \bar{7}X$

$9 = 8X$ $\qquad$ $\bar{9} = \bar{8}X$

We of course immediately get

$1 = 6A$ $\qquad$ $\bar{1} = \bar{6}A$

$6 = 7X^{-1}$ $\qquad$ $\bar{6} = \bar{7}X^{-1}$

$7 = 8X^{-1}$ $\qquad$ $\bar{7} = \bar{8}X^{-1}$

$8 = 9X^{-1}$ $\qquad$ $\bar{8} = \bar{9}X^{-1}$

We enter this information in tables I,II,III, and IV.

TABLE I

	X	X^{-1}	A	A^{-1}
1	1	1	2	6
2	3	5	3	1
3	4	2		2
4	5	3		
5	2	4		
6	7		1	
7	8	6		
8	9	7		
9		8		

TABLE II

	X	X^{-1}	A	A^{-1}
$\bar{1}$	$t_1\bar{1}$	$t_1^{-1}\bar{1}$	$\bar{2}$	$\bar{6}$
$\bar{2}$	$\bar{3}$	$t_2^{-1}\bar{5}$	$t_1^2\bar{3}$	$\bar{1}$
$\bar{3}$	$\bar{4}$	$\bar{2}$		$t_1^{-2}\bar{2}$
$\bar{4}$	$\bar{5}$	$\bar{3}$		
$\bar{5}$	$t_2\bar{2}$	$\bar{4}$		
$\bar{6}$	$\bar{7}$		$\bar{1}$	
$\bar{7}$	$\bar{8}$	$\bar{6}$		
$\bar{8}$	$\bar{9}$	$\bar{7}$		
$\bar{9}$		$\bar{8}$		

TABLE III

$t_1 = X$

1| 1   1|1

$t_2 = A\ X\ X\ X\ X\ A^{-1}$

1| 1   1|2|3|4|5|2|1

$t_3 = A^{-1}\ X\ X\ X\ X\ A$

1| 1   1| 6 |7|8|<u>9|6</u>|1   *

TABLE IV

	X	X	A	X	A^{-1}	A^{-1}	
1	1	1	2	3	2	1	
		6	1	1	6		
6	7	<u>8</u>	<u>5</u>	2	1	6	*
2	3	4			3	2	
5	2	3				5	
4	5	2	3	4		4	
3	4	5				3	
			4	5			
					5		
					4		
	6	7					
			6	7			
				6			
7	8	9				7	
			7	8			
					7		
8	9					8	
			8	9			
					8		
9						9	
			9				
					9		

Two new pieces of information occur, denoted by *. From TABLE III

$$9X = 6$$

$$6X^{-1} = 9$$

After rewriting the row we get

$$\overline{9}X = \overline{9}\ X^{-1}\ X^{-1}\ X^{-1}\ A\ t_3\ A^{-1}$$

$$= t_3\ \overline{6}$$

and $\overline{6}\ X^{-1} = t_3^{-1}\overline{9}$

Note that TABLE III is completely closed, hence it cannot provide any further information. We do not display it beyond this stage. Now from the row starting 6 in TABLE IV we see

$$8A = 5$$
$$5A^{-1} = 8$$

After rewriting the row we find

$$\overline{8}\,A = t_2^{-1}\,\overline{5}$$
$$\overline{5}\,A^{-1} = t_2\overline{8}$$

We enter this information in TABLE I and TABLE II and fill in TABLE IV wherever possible.

TABLE I

	X	X^{-1}	A	A^{-1}
1	1	1	2	6
2	3	5	3	1
3	4	2		2
4	5	3		
5	2	4		8
6	7	9	1	
7	8	6		
8	9	7	5	
9	6	8		

TABLE II

	X	X^{-1}	A	A^{-1}
$\overline{1}$	$t_1\overline{1}$	$t_1^{-1}\overline{1}$	$\overline{2}$	$\overline{6}$
$\overline{2}$	$\overline{3}$	$t_2^{-1}\overline{5}$	$t_1^{2}\overline{3}$	$\overline{1}$
$\overline{3}$	$\overline{4}$	$\overline{2}$		$t_1^{-2}\overline{2}$
$\overline{4}$	$\overline{5}$	$\overline{3}$		
$\overline{5}$	$t_2\overline{2}$	$\overline{4}$		$t_2\overline{8}$
$\overline{6}$	$\overline{7}$	$t_3^{-1}\overline{9}$	$\overline{1}$	
$\overline{7}$	$\overline{8}$	$\overline{6}$		
$\overline{8}$	$\overline{9}$	$\overline{7}$	$t_2^{-1}\overline{5}$	
$\overline{9}$	$t_3\overline{6}$	$\overline{8}$		

TABLE IV

	X	X	A	X	A^{-1}	A^{-1}	
1	1	1	2	3	2	1	
8	9	6	1	1	6	8	*
6	7	8	5	2	1	6	
2	3	4			3	2	
5	2	3				5	
4	5	2	3	4		4	
3	4	5				3	
			4	5	8		
					4		
9	6	7				9	
			6	7			
			9	6			
7	8	9				7	
			7	8			
					7		
			8	9			
					9		

From the row starting 8 (denoted by *) in TABLE IV we find

$$6A^{-1} = 8.$$

However when we go to enter this information in TABLE I we find

$$8A = 5.$$

We may conclude $6 = 5$.

We rewrite the row starting 8 in TABLE IV to get

$$\overline{6}A^{-1} = t_1^{-1}\, t_3^{-1}\, \overline{8}$$

or $$\overline{8}A = t_3\, t_1\, \overline{6}$$

From TABLE II we know

$$\overline{8}A = t_2^{-1}\overline{5}$$

so $$\overline{5} = t_2\, t_3\, t_1\, \overline{6}$$

We reproduce part of TABLE I so we can easily compare rows *5* and *6*

	X	X^{-1}	A	A^{-1}
5	*2*	*4*		*8*
6	*7*	*9*	*1*	

From the column headed by X we can conclude

1. *2* = *7*

(We will number the redundancies consecutively to aid in processing.)

From the column headed by X^{-1} we can conclude

2. *4* = *9*

There is also the new information

5A = *1*

We now multiply row $\overline{6}$ in TABLE II by $t_2 t_3 t_1$ and reproduce that part of TABLE II of interest

	X	X^{-1}	A	A^{-1}
$\overline{5}$	$t_2\overline{2}$	$\overline{4}$		$t_2\overline{8}$
$\overline{6}$	$t_2 t_3 t_1 \overline{7}$	$t_2 t_3 t_1 t_3^{-1}\overline{9}$	$t_2 t_3 t_1 \overline{1}$	

Corresponding to the equations obtained from TABLE I we can write down the following

$$t_2\overline{2} = t_2 t_3 t_1 \overline{7}$$

$$\overline{4} = t_2 t_3 t_1 t_3^{-1}\overline{9}$$

We also get the following new information which is entered in TABLE II

$$\bar{5}A = t_2 t_3 t_1 \bar{1}$$

We now erase row *6* in TABLE I; erase row $\bar{6}$ in TABLE II; replace occurrences of *6* by 5 in TABLE I; replace occurrences of $\bar{6}$ by $t_1^{-1} t_3^{-1} t_2^{-1} \bar{5}$ in TABLE II; and erase those rows in TABLE IV which contain *6*.

The tables for the example now look like:

TABLE I

	X	X^{-1}	A	A^{-1}
1	1	1	2	5
2	3	5	3	1
3	4	2		2
4	5	3		
5	2	4	1	8
7	8	5		
8	9	7	5	
9	5	8		

TABLE IV

	X	X	A	X	A^{-1}	A^{-1}	
1	1	1	2	3	2	1	
8	9	5	1	1	5	8	
5	2	3	5	2	1	5	*
2	3	4			3	2	
4	5	2	3	4		4	
3	4	5	1	1	5	3	*
			4	5	8		
					4		
7	8	9				7	
			7	8			
					7		
			8	9			
					9		

TABLE II

	X	X^{-1}	A	A^{-1}
$\bar{1}$	$t_1 \bar{1}$	$t_1^{-1} \bar{1}$	$\bar{2}$	$t_1^{-1} t_3^{-1} t_2^{-1} \bar{5}$
$\bar{2}$	$\bar{3}$	$t_2^{-1} \bar{5}$	$t_1^2 \bar{3}$	$\bar{1}$
$\bar{3}$	$\bar{4}$	$\bar{2}$		$t_1^{-2} \bar{2}$
$\bar{4}$	$\bar{5}$	$\bar{3}$		
$\bar{5}$	$t_2 \bar{2}$	$\bar{4}$	$t_2 t_3 t_1 \bar{1}$	$t_2 \bar{8}$
$\bar{7}$	$\bar{8}$	$t_1^{-1} t_3^{-1} t_2^{-1} \bar{5}$		
$\bar{8}$	$\bar{9}$	$\bar{7}$	$t_2^{-1} \bar{5}$	
$\bar{9}$	$t_3 t_1^{-1} t_3^{-1} t_2^{-1} \bar{5}$	$\bar{8}$		

The rows marked by an asterisk in TABLE IV contain new information. However we will delay dealing with this information until after all the current redundancies are processed. The next redundancy to be processed is number one.

1. $2 = 7$

$$\bar{2} = t_3\ t_1\ \bar{7}$$

We compare rows 2 and 7 in TABLE I to get the new redundancies

3. $3 = 8$

4. $5 = 5$

The corresponding columns in TABLE II yield (after multiplying row $\bar{7}$ by t_3t_1)

$$\bar{3} = t_3\ t_1\ \bar{8}$$

$$t_2^{-1}\ \bar{5} = t_3\ t_1\ t_1^{-1}\ t_3^{-1}\ t_2^{-1}\ \bar{5}$$

We replace all occurrences of 7 by 2 in TABLE I; replace all occurrences of $\bar{7}$ by $t_1^{-1}t_3^{-1}\bar{2}$ in TABLE II; and erase any rows containing 7 in TABLE IV. We now deal with the next redundancy, number two.

2. $4 = 9$

$$\bar{4} = t_2\ t_3\ t_1\ t_3^{-1}\ \bar{9}$$

We compare rows 4 and 9 in TABLE I to get the new redundancies

5. $5 = 5$

6. $3 = 8$

We multiply row $\bar{9}$ in TABLE II by $t_2t_3t_1t_3^{-1}$ and then compare with row $\bar{4}$ to obtain the corresponding equations

$$\bar{5} = t_2\ t_3\ t_1\ t_3^{-1}\ t_3\ t_1^{-1}\ t_3^{-1}\ t_2^{-1}\ \bar{5}$$

$$\bar{3} = t_2\ t_3\ t_1\ t_3^{-1}\ \bar{8}$$

We replace all occurrences of *9* by *4* in TABLE I; replace all occurrences of $\bar{9}$ by $t_3 t_1^{-1} t_3^{-1} t_2^{-1} \bar{4}$ in TABLE II; and erase any rows containing *9* in TABLE IV. The next redundancy to be dealt with is

3. *3* = *8*

$$\bar{3} = t_3\ t_1\ \bar{8}$$

We compare rows *3* and *8* TABLE I

7. *4* = *4* (*9* has been replaced by *4* under X)
8. *2* = *2* (*7* has been replaced by *2* under X^{-1})

We also get the new information

*3*A = *5*

which we put in TABLE I. We multiply row $\bar{8}$ in TABLE II by $t_3 t_1$ to get

$$\bar{4} = t_3\ t_1\ t_3\ t_1^{-1}\ t_3^{-1}\ t_2^{-1}\ \bar{4}$$

$$\bar{2} = t_3\ t_1\ t_1^{-1} t_3^{-1}\ \bar{2}$$

and the equation for the new information is

$$\bar{3}A = t_3\ t_1\ t_2^{-1}\ \bar{5}$$

which we put in TABLE II. We now replace *8* by *3* in TABLE I ; replace $\bar{8}$ by $t_1^{-1} t_3^{-1} \bar{3}$ in TABLE II; erase any row in TABLE IV which contains *8*; and in redundancy number 6, *8* is replaced by *3* and $\bar{8}$ is replaced by $t_1^{-1} t_3^{-1} \bar{3}$. The next redundancy to be processed is

4. *5* = *5*

$$t_2^{-1} \bar{5} = t_3\ t_1\ t_1^{-1}\ t_3^{-1}\ t_2^{-1}\ \bar{5}$$

When we simplify we find

$$t_2^{-1} \bar{5} = t_2^{-1} \bar{5}$$

which is a trivial relation and is ignored.

5. $5 = 5$

$$\overline{5} = t_2\ t_3\ t_1\ t_3^{-1}\ t_3\ t_1^{-1}\ t_3^{-1}\ t_2^{-1}\ \overline{5}$$

This is again a trivial relation and ignored.

6. $3 = 3$

$$\overline{3} = t_2\ t_3\ t_1\ t_3^{-1}\ t_1^{-1}\ t_3^{-1}\ \overline{3}$$

We cancel $\overline{3}$ to get the following relation

$$\text{A.}\quad e = t_2\ t_3\ t_1\ t_3^{-1}\ t_1^{-1}\ t_3^{-1}$$

which we record as relation **A**.

7. $4 = 4$

$$\overline{4} = t_3\ t_1\ t_3\ t_1^{-1}\ t_3^{-1}\ t_2^{-1}\ \overline{4}$$

This relation is the inverse of relation A above and so is already recorded.

8. $2 = 2$

$$\overline{2} = t_3\ t_1\ t_1^{-1}\ t_3^{-1}\ \overline{2}$$

This again leads to a trivial relation and so is ignored.

We have now dealt with all the redundancies. All new information has been entered and processed. On the next page we give the tables as they now appear.

TABLE I

	X	X^{-1}	A	A^{-1}
1	1	1	2	5
2	3	5	3	1
3	4	2	5	2
4	5	3		
5	2	4	1	3

TABLE IV

	X	X	A	X	A^{-1}	A^{-1}	
1	1	1	2	3	2	1	
2	3	4	4	5	3	2	*
3	4	5	1	1	5	3	
4	5	2	3	4		4	
5	2	3	5	2	1	5	
					4		

TABLE II

	X	X^{-1}	A	A^{-1}
$\overline{1}$	$t_1\overline{1}$	$t_1^{-1}\overline{1}$	$\overline{2}$	$t_1^{-1}t_3^{-1}t_2^{-1}\overline{5}$
$\overline{2}$	$\overline{3}$	$t_2^{-1}\overline{5}$	$t_1^{2}\overline{3}$	$\overline{1}$
$\overline{3}$	$\overline{4}$	$\overline{2}$	$t_3 t_1 t_2^{-1}\overline{5}$	$t_1^{-2}\overline{2}$
$\overline{4}$	$\overline{5}$	$\overline{3}$		
$\overline{5}$	$t_2\overline{2}$	$\overline{4}$	$t_2 t_3 t_1\overline{1}$	$t_2 t_1^{-1} t_3^{-1}\overline{3}$

The only blank spaces remaining are in line 4 of TABLE I and correspondingly in line $\overline{4}$ of TABLE II. How ever the line beginning with 2 in TABLE IV (denoted by *) gives

$$4A = 4$$

$$\overline{4}A = t_1^2\ t_3\ t_1\ t_2^{-1}\ \overline{4}$$

We enter this information in tables I and II. We rewrite the rows beginning 3, 4, and 5 in TABLE IV. These rows are complete because of information gained from the redundancies. We use this information to rewrite the rows and complete the algorithm.

$$\overline{3} = \overline{3}\ X\ X\ A\ X\ A^{-1}\ A^{-1}$$

$$\bar{3} = \bar{4}\, X\, A\, X\, A^{-1}\, A^{-1}$$

$$= \cdots = t_2\, t_3\, t_1\, t_3^{-1}\, t_1^{-1}\, t_3^{-1}\, \bar{3}$$

This leads to relation A which we already have.

$$\bar{4} = \bar{4} X\, X\, A\, X\, A^{-1}\, A^{-1}$$

$$\vdots$$

$$= t_2\, t_1^2\, (t_2\, t_1^{-1}\, t_3^{-1}\, t_1^{-1})^2\, \bar{4}$$

or $$e = t_2\, t_1^2\, (t_2\, t_1^{-1}\, t_3^{-1}\, t_1^{-1})^2$$

Let us call this relation B.

$$\bar{5} = \bar{5}\, X\, X\, A\, X\, A^{-1}\, A^{-1}$$

$$\vdots$$

$$= t_2\, t_3\, t_1\, t_2^{-1}\, t_2\, t_1^{-1}\, t_3^{-1}\, t_2^{-1}\, \bar{5}$$

$$= \bar{5}$$

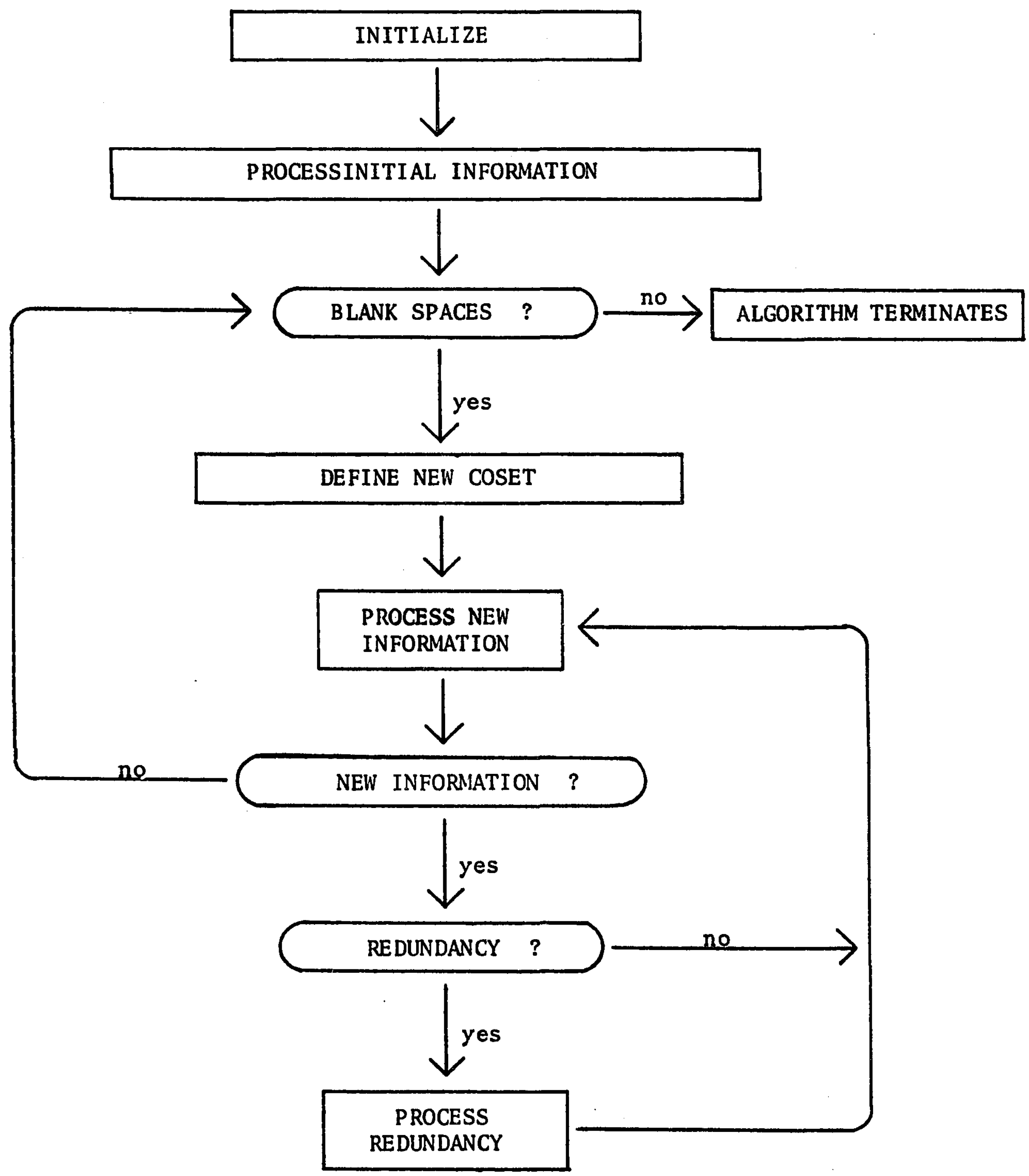
When we collect all the relations which we have derived we have a presentation for our subgroup H;

$$H = \langle\, t_1,\, t_2,\, t_3\, ;\ A,\, B\, \rangle$$

The description of the algorithm is complete however a few general remarks may be of some help. At any time in the execution of the algorithm we are in one of three different states. The first state is that there are blank spaces in TABLE I but all information has been processed. In this case new cosets are defined. The second state is that there are blank spaces in TABLE I and there is information to be processed (redundancies etc.). The third state is terminal when there are no blank spaces in TABLE I. In this case for any coset defined we rewrite each row in TABLE IV not already rewritten in order to get a complete set of defining relations. This is the final step.

It has been proved that if the index of the subgroup is finite the algorithm will terminate (Mendelsohn, 1964 & 1965). However it is impossible to say the number of cosets needed to be defined before termination of the algorithm. This is equivalent to the solution of the classical word problem and known to be unsolvable.

FLOW CHART FOR ALGORITHM



CHAPTER TWO

In this chapter we would like to discuss the classical Reidemeister-Schreier method for presenting a subgroup. We do not give proofs for the various assertions since these proofs are easily obtainable (Hall 1959; Magnus,Karrass & Solitar 1966). However we will give a summary of the algorithm so the reader may compare with chapter one.

The rewriting process was mentioned in the previous chapter. This process is similar in the Reidemeister-Schreier algorithm. Specifically let G be a group with the presentation

$$G = \langle a_1, a_2, \ldots a_n ; R_1(a), \ldots \rangle$$

and let H be a subgroup generated by the words $J_i(a), \ldots$. Then a rewriting process is one which takes a word in H written in the symbols a_k and gives as a result the same word but written in the symbols J_i. Let us agree to rename $J_i(a)$ by s_i and consider the following:

Theorem 2.1 Suppose τ is a rewriting process for H then a presentation for H is obtained by using the symbols s_i as generating symbols and using the following equations as defining relations;

(1) $s_i = \tau(J_i(a))$

(2) $\tau(V) = \tau(V_1)$

where V and V_1 are words in H which differ only by the insertion or deletion of trivial relators.

(3) $\tau(V_1 V_2) = \tau(V_1)\tau(V_2)$

where V_1 and V_2 define elements of H.

(4) $\tau(WR_kW^{-1}) = e$

where R_k is a defining relator for G and W is any word in G.
####

This theorem is used to prove further results but has no practical significance. The reason this theorem has no practical application is the considerable simplification that can be made by a particular choice of generators and rewriting process. One such choice involves right cosets. So let

$U = \{ e, u_1, u_2, \ldots \}$

be a set of right coset representatives for H. Further if g is a word in G then let

$\bar{g} = u_i$

where $Hg = Hu_i$

i.e. $\bar{\ }$ is a mapping from G to U. Now by a theorem of Schreier it is possible to choose the set U so that it satisfies the following:

If $u_j = a_1a_2\ldots a_t$ and u_j is in U then $a_1a_2\ldots a_{t-1}$ is also in U. The system of coset representatives is called a Schreier system and the property above is called the Schreier property.

Theorem 2.2 H is generated by

$\{ ua_i\overline{ua_i}^{\,-1}$: u is in U, a_i generator for G $\}$
####

It is easy to see that a word $ua\overline{ua}^{\,-1}$ is in H however to show every word in H is a product of elements of that form is somewhat technical. We have obtained generators for H so we now need a rewriting process.When we rewrite using a Schreier system and a rewriting process called a Reidemeister rewriting process we end up with a Reidemeister-Schreier rewriting process. Let us use the symbol $S_{u,a}$ to represent $ua\overline{ua}^{-1}$. Suppose we have a word V in H

$$V = a_r^\circ a_s^\circ \dots a_w^\circ \qquad \circ = \pm 1$$

Then if τ is a Reidemeister-Schreier rewriting process

$$\tau(V) = S^\circ_{u_r,a_r} S^\circ_{u_s,a_s} \dots S^\circ_{u_w,a_w}$$

where $\circ = \pm 1$ according as it is +1 or -1 in V

and u_t is the coset representative of the initial segment of V preceding a_t if a_t has exponent +1 or u_t is the representative of the initial segment of V up to and including a_t^{-1} if a_t has exponent -1.

e.g. $u_r = e$ if $V = a_r a_s^\circ \dots a_w^\circ$

$u_r = \overline{a_r^{-1}}$ if $V = a_r^{-1} a_s^\circ \dots a_w^\circ$

e.g. Suppose $a_1^2 a_2^{-1} a_3$ defines an element of H.

$$\tau(a_1^2 a_2^{-1} a_3) = S_{e,a_1} S_{\bar{a}_1,a_1} S^{-1}_{\overline{a_1 a_1 a_2^{-1}},a_2} S_{\overline{a_1,a_1,a_2^{-1}},a_3}$$

We now state our final theorem:

Theorem 2.3 If τ is a Reidemeister- Schreier rewriter then

$$H = \langle\, S_{u,a_r}, \dots ; S_{M,a_s}, \dots, \tau(uR_i u^{-1}), \dots \,\rangle$$

where: u is an arbitrary Schreier coset representative

a_r is an arbitrary generator

R_i is an arbitrary defining relator for G

and M is a Schreier representative and a_s a generator such that Ma_s is the same as $\overline{Ma_s}$ except for the addition or deletion of some trivial relators.

####

In order to show the details of the process we work out the same example as in chapter one.

$G = \langle A, X ; X^2AX = A^2 \rangle$

$H = \{ t_1 = X, t_2 = AX^4A^{-1}, t_3 = A^{-1}X^4A \}$

From chapter one we know the index of H in G is 5 and the following are a Schreier system of coset representatives.

e, A, AX, AX^2, AX^3

We also know the values in TABLE I and so we can find $\overline{W}$ where W is any word in G. From Theorem 2.3, H is generated by the symbols:

$S_{e,A}, S_{A,A}, S_{AX,A}, S_{AX^2,A}, S_{AX^3,A}$

$S_{e,X}, S_{A,X}, S_{AX,X}, S_{AX^2,X}, S_{AX^3,X}$

i.e. for every coset representative and every generator for G we have a generator for H. Since there are 5 cosets and 2 generators for G we have $2 \cdot 5 = 10$ generators for H.

From Theorem 2.3, H has two kinds of relators. The first kind gives us the following as relators:

$S_{e,A}, S_{A,X}, S_{AX,X}, S_{AX^2,X}$

i.e. $S_{AX,X}$ is a relator because

$AX \cdot X = AX^2$

and $\overline{AX \cdot X} = AX^2$.

There are 5 relators of the second kind. We let τ be the Reidemeister-Schreier rewriting process, and then the 5 relators are

1. $\tau(X^2AXA^{-2})$
2. $\tau(AX^2AXA^{-3})$

3. $\tau(AX^3AXA^{-2}X^{-1}A^{-1})$

4. $\tau(AX^4AXA^{-2}X^{-2}A^{-1})$

5. $\tau(AX^5AXA^{-2}X^{-3}A^{-1})$

i.e. We have $\tau(URU^{-1})$ where U is a coset representative and R is a relator. There are 5 coset representatives and 1 relator so we have $5\cdot 1 = 5$ relators of the second kind. We now rewrite each of the above.

1. $\tau(XXAXA^{-1}A^{-1})$

$$= S_{e,X}\, S_{\overline{X},X}\, S_{\overline{XX},A}\, S_{\overline{XXA},X}\, S^{-1}_{\overline{XXAXA^{-1}},A}\, S^{-1}_{\overline{XXAXA^{-1}A^{-1}},A}$$

$$= S_{e,X}\, S_{e,X}\, S_{e,A}\, S_{A,X}\, S^{-1}_{A,A}\, S^{-1}_{e,A}$$

2. $\tau(AXXAXA^{-1}A^{-1}A^{-1})$

$$= S_{e,A}\, S_{A,X}\, S_{AX,X}\, S_{AX^2,A}\, S_{AX^2,X}\, S^{-1}_{AX,A}\, S^{-1}_{A,A}\, S^{-1}_{e,A}$$

3. $\tau(AXXXAXA^{-1}A^{-1}X^{-1}A^{-1})$

$$= S_{e,A}\, S_{A,X}\, S_{AX,X}\, S_{AX^2,X}\, S_{AX^3,A}\, S_{e,A}\, S^{-1}_{AX^3,A}\, S^{-1}_{AX,A}\, S^{-1}_{A,X}\, S^{-1}_{e,A}$$

4. $\tau(AXXXXAXA^{-1}A^{-1}X^{-1}X^{-1}A^{-1})$

$$= S_{e,A}\, S_{A,X}\, S_{AX,X}\, S_{AX^2,X}\, S_{AX^3,X}\, S_{A,A}\, S_{AX,X}\, S^{-1}_{AX^2,A}\, S^{-1}_{AX^2,A}\, S^{-1}_{AX,X}\, S^{-1}_{A,X}\, S^{-1}_{e,A}$$

5. $\tau(AXXXXXAXA^{-1}A^{-1}X^{-1}X^{-1}X^{-1}A^{-1})$

$$= S_{e,A}\, S_{A,X}\, S_{AX,X}\, S_{AX^2,X}\, S_{AX^3,X}\, S_{A,X}\, S_{AX,A}\, S_{AX^3,X}\, S^{-1}_{e,A}\, S^{-1}_{AX^3,A}\, S^{-1}_{AX^2,X}\, S^{-1}_{AX,X}\, S^{-1}_{A,X}\, S^{-1}_{e,A}$$

We now delete the relators of the first kind from the generating set and from the remaining relators. We are left with the generators:

$$S_{A,A},\ S_{AX,A},\ S_{AX^2,A},\ S_{AX^3,A},\ S_{e,X},\ S_{AX^3,X}$$

and the following relators:

1. $S^2_{e,X}\, S^{-1}_{A,A}$

2. $S_{AX^2,A}\, S^{-1}_{AX,A}\, S^{-1}_{A,A}$

3. $S_{AX^3,A}\, S_{e,X}\, S^{-1}_{AX^3,A}\, S^{-1}_{AX,A}$

4. $S_{AX^3,X}\, S_{A,A}\, S^{-2}_{AX^2,A}$

5. $S_{AX^3,X}\, S_{AX,A}\, S_{AX^3,X}\, S^{-1}_{AX^3,A}$

We now use the 5 relators to eliminate all generators except $S_{AX^3,A}$ and $S_{AX,A}$.

From 5. $\quad S_{AX^3,A} = S_{AX^3,X}\, S_{AX,A}\, S_{AX^3,X}$

From 3. $\quad S_{e,X} = S^{-1}_{AX^3,A}\, S_{AX,A}\, S_{AX^3,A}$

$$= S^{-1}_{AX^3,X}\, S^{-1}_{AX,A}\, S^{-1}_{AX^3,X}\, S_{AX,A}\, S_{AX^3,X}\, S_{AX,A}\, S_{AX^3,X}$$

From 1. $\quad S_{A,A} = S^2_{e,X}$

$$= (\, S^{-1}_{AX^3,X}\, S^{-1}_{AX,A}\, S^{-1}_{AX^3,X}\, S_{AX,A}\, S_{AX^3,X}\, S_{AX,A}\, S_{AX^3,X}\,)^2$$

From 2. $\quad S_{AX^2,A} = S_{A,A}\, S_{AX,A}$

$$= (\, S^{-1}_{AX^3,X}\, S^{-1}_{AX,A}\, S^{-1}_{AX^3,X}\, S_{AX,A}\, S_{AX^3,X}\, S_{AX,A}\, S_{AX^3,X}\,)^2 \cdot S_{AX,A}$$

We have written every generator in terms of $S_{AX^3,X}$ and $S_{AX,A}$ hence we may delete from the presentation every generator except $S_{AX^3,X}$ and $S_{AX,A}$. We are left only with relator 4 which we rewrite in terms of $S_{AX^3,X}$ and $S_{AX,A}$. Let

$$B = S_{AX^3,X}$$

$$C = S_{AX,A}$$

Then we may write 4 as:

$$4.\ B(B^{-1}C^{-1}B^{-1}CBCB)(B^{-1}C^{-1}B^{-1}CBCB)C^{-1}(B^{-1}C^{-1}B^{-1}CBCB)(B^{-1}C^{-1}\cdot B^{-1}CBCB)C^{-1}(BC^{-1}B^{-1}CBCB)(B^{-1}C^{-1}B^{-1}CBCB)$$
$$= (C^{-1}B^{-1}C^{2}BCB)(C^{-1}B^{-1}C^{-1}B^{-1}C^{-2}BCB)^{2}$$
$$= BCBC^{-1}B^{-1}C^{2}(BCBC^{-1}B^{-1}C^{-1}B^{-1}C^{-2})^{2}$$

Thus

$$H = \langle\ B, C\ ;\ 4\ \rangle .$$

If we compare this with chapter one we see the two presentations are nearly the same. They can be made exactly the same if we eliminate the generator t_2 from the presentation in chapter one. Thus the methods of chapter one and chapter two give the same results in this case. In the next chapter we prove that this is always the case under certain conditions.

CHAPTER THREE

In this chapter we will prove that the extended Todd-Coxeter algorithm of chapter one does give a presentation for the subgroup. We should keep in mind the restrictions that the group G must be finitely presented and that the index of the subgroup H in G must be finite. Let us assume we have a group G

$$G = \langle a_1, a_2, \ldots a_m ; R_1(a), R_2(a), \ldots R_g(a) \rangle$$

and we have a subgroup H generated by n words

$$t_k = t_k(a) \qquad k = 1, 2, \ldots n.$$

We assume that the index of H in G is finite and we have completed a coset enumeration. Thus we have the tables I-IV of chapter one and also a number of relators derived from the coset enumeration. We will use Theorem 2.1 to prove that these relations are sufficient to present H in conjunction with the generators t_k. Now in order to use Theorem 2.1 we must first prove we have a rewriting process. Actually the process we use does more than a rewriting process. For suppose we have a word W in G

$$W = a_i a_j \ldots a_r.$$

We multiply W by $\mathcal{T}$ and use TABLE II to rewrite W. Let us denote this process by τ'

e.g. $\qquad \tau'(W) = \mathcal{T} a_i a_j \ldots a_r$

If we look in TABLE II we might see

$$\mathcal{T} a_i = \alpha_i \mathcal{I}$$

so $\qquad \tau'(W) = \alpha_i \mathcal{I} a_j \ldots a_r.$

We continue in this manner until we have an equation of the form

$$\tau'(W) = \alpha_i\alpha_j \ldots \alpha_r \bar{\pi}$$

where $\alpha_i \ldots \alpha_r$ are words in H. Now we know

$$\tau'(W) = W$$

so $$W = \alpha_i\alpha_j \ldots \alpha_r \bar{\pi}.$$

If $\bar{\pi} = \bar{T}$, then W is a word in H and

$$W = \alpha_i\alpha_j \ldots \alpha_r.$$

So we have expressed W in terms of words in H. If $\bar{\pi} \neq \bar{T}$ then W is not a word in H. Our processs not only rewrites a word but also decides whether or not that word is in the subgroup. Now according to the definition in chapter two a rewriting process takes any word in H written in " a " symbols and gives the same word but written in " t " symbols. We see this is exactly what the above process does so it is a rewriting process. Let us recall Theorem 2.1. It said:

> If τ is a rewriting process, then the following relations are sufficient to present H.
>
> 1. $s_i = \tau(t_i(a))$
> 2. $\tau(V) = \tau(V_1)$ where V and V_1 differ by trivial relators.
> 3. $\tau(V_1V_2) = \tau(V_1)\tau(V_2)$
> 4. $\tau(WR_jW^{-1}) = e$

We must show that the relations we have derived from the coset enumeration enable us to prove the above relations and then our set of relations will also be sufficient to present H.

1. $s_i = \tau(t_i(a))$ In order to show this relation is implied by

the relations of the extended coset enumeration we merely need to say what this relation means. It means we rewrite the generators of H and set each generator equal to a new symbol. However TABLE III in the coset enumeration is precisely this operation. Thus the relations

$$s_i = \tau(t_i(a))$$

are incorporated in the coset enumeration for all i.

2. $\tau(V) = \tau(V_1)$ Now V and V_1 differ by a trivial relator so let

$$V = a_1^\circ a_2^\circ \ldots a_i^\circ a_j^\circ \ldots a_v^\circ \qquad \circ = \pm 1$$

$$V_1 = a_1^\circ a_2^\circ \ldots a_i^\circ a_p a_p^{-1} a_j^\circ \ldots a_v^\circ \qquad \circ = \pm 1$$

When we apply the rewriting process τ' it will certainly give the same result up to and including a_i°. We may suppose

$$\tau'(V) = \alpha_1^\circ \alpha_2^\circ \ldots \alpha_i^\circ \, \mathcal{I} \, a_j^\circ \ldots a_v^\circ$$

$$\tau'(V_1) = \alpha_1^\circ \alpha_2^\circ \ldots \alpha_i^\circ \, \mathcal{I} \, a_p a_p^{-1} a_j^\circ \ldots a_v^\circ$$

where $\alpha_1, \alpha_2, \ldots \alpha_i$ are words in H. Now we look in TABLE II and let

$$\mathcal{I} \, a_p = \beta_p \, \overline{p}.$$

Thus $$\tau'(V_1) = \alpha_1^\circ \alpha_2^\circ \ldots \alpha_i^\circ \, \beta_p \, \overline{p} \, a_p^{-1} a_j^\circ \ldots a_v^\circ$$

We have from above

$$\overline{p} \, a_p^{-1} = \beta_p^{-1} \, \mathcal{I}$$

and substituting this in the above expression we see

$$\tau'(V_1) = \alpha_1^\circ \alpha_2^\circ \ldots \alpha_i^\circ \, \beta_p \beta_p^{-1} \, \mathcal{I} \, a_j^\circ \ldots a_v^\circ$$

$$= \alpha_1^\circ \alpha_2^\circ \ldots \alpha_i^\circ \, \mathcal{I} \, a_j^\circ \ldots a_v^\circ$$

which is exactly the same as $\tau'(V)$. Thus $\tau'(V_1) = \tau'(V)$ and relation 2 is derivable .

3. $\tau(V_1V_2) = \tau(V_1)\tau(V_2)$ To prove this relation is to prove τ' is finitely distributive over the words in H. However we know this is true by the definition of τ'. The application of τ' to a word consists of multiplying the first symbol in the word by $\mathcal{T}$ and then carrying out multiplication symbol by symbol. Thus if $\mathcal{T}$ occurs to the left of any symbol we may consider that as a new application of τ'. We are given that V_1 is a word in H so let

$$\tau'(V_1) = \alpha \mathcal{T}$$

Then it is easy to see

$$\begin{aligned}\tau'(V_1V_2) &= \alpha \mathcal{T} V_2 \\ &= \alpha\, \tau'(V_2) \\ &= \tau'(V_1)\, \tau'(V_2).\end{aligned}$$

4. $\tau(WR_jW^{-1}) = e$ To show this we recall how the relations we derived from the coset enumeration arose. For every relation R_j in G and every coset i we applied $\mathcal{I}$ to the relation and got an equation

$$\mathcal{I}R_j = S\mathcal{I}$$
$$\mathcal{I}R_j\mathcal{I}^{-1} = S$$

Thus S would be a relation in the subgroup. Now let us rewrite W

$$\tau'(W) = \alpha\mathcal{L}$$

where α is a word in H and $\mathcal{L}$ is the coset representative of the coset to which W belongs. We have of course

$$W = \alpha\mathcal{L}$$
$$W^{-1} = \mathcal{L}^{-1}\alpha^{-1}$$

We substitute this expression for W in 4

$$\tau'(WR_jW^{-1}) = \tau'(\alpha\mathcal{L}R_j\mathcal{L}^{-1}\alpha^{-1})$$

$$= \tau'(\alpha)\ \tau'(\mathcal{L}R_j\mathcal{L}^{-1})\ \tau'(\alpha^{-1})$$

where we may distribute τ' by relation 3 (already proved). Now $\tau'(\mathcal{L}R_j\mathcal{L}^{-1})$ is a relator we derived in the coset enumeration and so may be deleted. We have now

$$\tau'(WR_jW^{-1}) = \tau'(\alpha)\ \tau'(\alpha^{-1})$$

$$= \tau'(\alpha\alpha^{-1})$$

$$= \tau'(e)$$

$$= e$$

Thus every relation of the form $\tau(WR_jW^{-1}) = e$ can be derived from the relations obtained by the coset enumeration technique. We have now proved that the extended coset enumeration technique gives a presentation for H.

At this point, we would like to compare the extended coset enumeration method with other methods available. We have sketched the classical Reidemeister-Schreier method and the advantages of coset enumeration are easy to see. First, in order to use the Reidemeister-Schreier algorithm it is necessary to perform an ordinary coset enumeration. After getting the coset representatives U, it is necessary to rewrite the products (UR_jU^{-1}) in terms of Schreier generators. In the extended version of coset enumeration we just do one coset enumeration and the process is finished. All rewriting is included in the coset enumeration. Further the presentation we end up with is in terms of the original generators not Schreier generators as in the Reidemeister-Schreier algorithm. Also in the latter case the number of subgroup generators is proportional to the number of cosets while in the coset enumeration technique the number of generators is fixed. Thus there are generally a large number of redundant Schreier generators, hence there must be an equally large number of relators to be able to eliminate the redundant generators. This is of course extra work when compared to coset enumeration. We are justified in concluding that coset enumeration is superior to the Reidemeister-Schreier algorithm.

There are two other methods published which attempt to solve some of the problems we have solved. These are due to Mendelsohn (1967), Mendelsohn and Benson (1966), and Leech(1962). The method of Leech is fairly complicated as he uses coset enumeration to prove group identities. However in his method one must record separately all informationwhich occurs in row closure. Then to prove an identity he rewrites the identity but using row closure information to simplify until only the identity remains. Further if there are redundancies in the initial enumeration, he must use the redundancies to get further relations until a set of relations has been found with which a coset enumeration can avoid redundancies and the original algorithm is then applicable. In the coset enumeration we have described, the information which occurs in row closure is automatically integrated and so there is no necessity to keep a separate record. Thus when we rewrite a word the row closure information is used automatically and no separate reference is needed. This is the advantage extended coset enumeration enjoys over Leech's method.

Mendelsohn has published several papers and a computer program describing various algorithms similar to the one we have described. In his earlier papers (1964,1965) he describes an algorithm which will write any word in G as a word in the subgroup H times a coset representative. Together with Benson, Mendelsohn made available a program which will perform the above function. His latest statement (1967) is a solution to the problem we solved in chapter one, namely find a presentation for a subgroup of a group in terms of subgroup generators. However his solution is very cumbersome. He claims it is still necessary to use Reidemeister-Schreier generators in an intermediate step. However we have proved in this chapter that this is not the case. In his paper (1967) he gives an example which purports to show the necessity of using Reidemeister-Schreier generators. This example is given in chapter five and a presentation is obtained without the use of special generators. Thus, using his method it is necessary to introduce the Reidemester-Schreier generators however the method made available in this thesis is a step forward in that it is not necessary to introduce these generators but merely proceed with the original subgroup generators.

CHAPTER FOUR

In this chapter we would like to discuss coset enumeration by computer. There have been a number of programs written which perform coset enumeration on a machine. Leech (1963) gives a good account of the work done up to 1963. The first computer methods used were in 1953! Since that time there have been many improvements in the methods plus considerable advances in hardware. One program due to Trotter (1964, 1966) will enumerate over 500 cosets in less than 10 seconds. Even faster programs are now available (M. Guy, unpublished). Coset enumeration of over 100,000 cosets has been accomplished. However there has been only one attempt to produce a program similar to the extended coset enumeration (Mendelsohn and Benson, 1966). This procedure is of course much more difficult and time consuming. Some of the difficulties that arise will be mentioned later.

The program that we wrote follows the outline given in chapter one quite closely. The logic is the same. However, there are many obscuring details due to the necessity of keeping track of minor details. For example it is necessary to keep a list of those rows which have already been rewritten, so the machine won't do them again. This list has to be checked every time the machine starts doing a row. There is one mainprogram and ten subprograms. We will explain in detail what is the function of each part.

1. Mainprogram: This program has three functions. The first is to input and initialize the group data. The second is an overall

control of execution i.e. it directs the machine to the various subprograms. The last function is to output the data. We would like to go through the program and show each section in detail. The first section statements 1-7, initializes the computer. Common areas are set aside, dimensions created and switches are set. For example a switch labelled " AMT " determines the number of redundancies waiting to be processed. Since there are no redundancies at the beginning AMT is set equal to 0. The next part of the first section is where the data is inputted. Note that all data to be inputted is numeric and all output is numeric. The data to be read in includes the length of each subgroup generator (expanded) in terms of group generators, the length of each group relator, the subgroup generators and the relators. The next initialization to be performed is to set two arrays " CTIB " and " WTAB " equal to zero. Now WTAB corresponds to TABLE I and CTIB corresponds to TABLE II. The machine searches these tables constantly and so every entry must be defined. When an entry is zero it means there is no information. Initially we have no information so every entry is made equal to zero. The last array to be initialized is called " NREL ". This array keeps track of which rows in each relator are complete. If a row is complete the entry is 0. We initialize NREL at zero except for the first row which we set at 1. Whenever a new coset is defined, a nonzero row is made in NREL corresponding to the number of the new coset.

The second section of the program is the control function. It is very simple. It first tells the machine to run through all the rows in tables III and IV (subroutine RUNTRU). If both these tables are filled then we go to the output section. If there are still blank spaces the program checks for redundancies. If there are some redundancies, subroutine RDUN2 is called to deal with them. If there are no redundancies new cosets are defined (subroutine DEFINE) and the program goes back to subroutine RUNTRU. This sequence continues until there are no blank spaces and we get to the output section.

The output section prints out the information we want. The first output is the number of cosets defined. This program does not have a consolidation routine hence it is also necessary to print out those cosets which are redundant. Finally the index is printed. This is of course the number of cosets defined minus the number of redundancies. The next output is TABLE I. The rows are numbered consecutively. If any row is zero it means that number was a redundant coset. The final output is TABLE II. However since the entries to TABLE II can be arbitrarily long we only print reference numbers in the body of the table. That is, suppose we want the result of coset i times generator j. The entry may be 0 which means coset $i \times$ generator j is simply equal to another coset. On the other hand there may be a nonzero number printed. This number is a reference number. To find the actual entry one looks in the list printed below until this reference number is found. The actual word is printed immediately below. The output is numeric. If we had 3 subgroup generators A,B,C then A is outputted as 1, B as 2, C as 3, A^{-1} as 4, B^{-1} as 5, C^{-1} as 6. So if a word is given as

15345

we would read it as $AB^{-1}CA^{-1}B^{-1}$.

2. Block Data: This is a technical FORTRAN subprogram to input data without a direct readin. We set the values of various group parameters.

3. CALC: This subroutine solves for new information. That is suppose we have found a row which gives new information. This subroutine solves for that new information and puts it in tables I and II. It makes the distinction between rewriting a subgroup generator and a relator.

4. RDUN2: This subroutine deals with redundancies. The information that is inputted has the following form

$$W\bar{\mathcal{I}} = V\bar{\mathcal{J}}$$

where W and V are words in the subgroup and $\bar{\mathcal{I}}$ and $\bar{\mathcal{J}}$ are coset representatives. The subroutine performs exactly those actions outlined in chapter one necessary to deal with redundancies.

i) Solves for the smaller number in terms of the larger.

ii) Compares rows in tables I and II.

iii) Eliminates all occurrences of the larger number.

The subroutine then checks to see if there are further redundancies in which case it deals with them, otherwise it returns control to the main-program.

5. RUNTRU: This subroutine checks each row in TABLE III and in TABLE IV for new information. It first checks TABLE III. To check a row the program starts to rewrite from the beginning of the row. It goes forward as far as possible. It then starts at the end and goes backward.

If the two directions just meet then there is new information and control is passed to subroutine CALC. If the machine fails to complete the row it goes on to the next row. If a redundancy occurs, this redundancy is put in an array to await processing. The subroutine checks through TABLE III first and then TABLE IV. The routine also zeroes out any row which is complete.

6. DEFINE: This subroutine has a very simple function. It merely scans the rows in tables III and IV until it comes to a blank space and defines a new coset at that blank space. It checks the subgroup generators first. If there is a blank space a new coset is defined, but if not then TABLE III is zeroed out entirely since there can be no new information there. The subroutine then starts checking TABLE IV for blank spaces. One interesting feature of the program is the reuse of redundant cosets. When a coset has been found to be redundant, it may then be redefined by this subroutine. That is, the number is reused.

7. NSUM: This subroutine is called whenever two words in the subgroup are to be multiplied together. The subroutine creates new space for the resulting word.

8. CHECK: This subroutine suppresses trivial relators of the form XX^{-1}. It is called after any word is made. e.g. CHECK is called by NSUM.

9. INV: This subroutine gives the inverse of a word inputted. The inverse operation is done using modular arithmetic. For example if we have 3 subgroup generators A,B,C inputted as 1,2,3 then a word B^2AC^{-2}

would be outputted as

22166

and the inverse would be

33455

i.e. we add 3 to each number mod 6 and then invert.

10. DOD: This subroutine performs modular arithmetic. The input is two numbers x,y and the output is the value of x mod y. The reason that the library subprograms are not used is that all arithmetic in this program is using halfwords of storage instead of full words. This results in storage efficiency but means we cannot use the library subprograms.

We reproduce a sample output below. The actual output is contained in a flap at the end of the thesis. The group we use as an example is the same group we used in chapters one and two. The program does not output the actual result of rewriting the rows but merely lists tables I and II. We see that if we make the translation

$1 \to t_1 \qquad 4 \to t_1^{-1}$

$2 \to t_2 \qquad 5 \to t_2^{-1}$

$3 \to t_3 \qquad 6 \to t_3^{-1}$

that these tables are almost exactly the same as in chapter one. The only difference is that occurring in TABLE II. The difference is due to the processing order of the computer. If we use the relation

$$t_2 t_3 t_1 t_3^{-1} t_1^{-1} t_3^{-1}$$

to rearrange some of the entries then there will be no difference.

The important part of the output is as follows:

NUMBER OF COSETS DEFINED 9

REDUNDANT COSETS 6 9 8 7

THE INDEX OF THE SUBGROUP IS

5

WTAB-MULTIPLICATION TABLE FOR COSETS

1:	1	2	1	5
2:	3	3	5	1
3:	4	5	2	2
4:	5	4	3	4
5:	2	1	4	3
6:	0	0	0	0
7:	0	0	0	0
8:	0	0	0	0
9:	0	0	0	0

CTIB-POINTERS TO LOCATION OF COSET REPRESENTATIVES

1:	1	0	3	48
2:	0	8	13	0
3:	0	79	0	5
4:	0	153	0	161
5:	11	30	0	97
6:	0	0	0	0
7:	0	0	0	0
8:	0	0	0	0
9:	0	0	0	0

THE ENTRIES TO TABLE II WHERE THE FIRST NUMBER IS THE LOCATION NUMBER IN THE TABLE ABOVE, THE SECOND NUMBER IS ONE MORE THAN THE NUMBER OF SYMBOLS IN THE WORD AND THEN THE WORD IS PRINTED BELOW IN NUMERIC FORM. EACH NUMBER REPRESENTS A SUBGROUP GENERATOR.

1	2 1
3	2 4
48	4 4 6 5
8	3 1 1
13	2 5
79	6 2 3 1 6 5
5	3 4 4
153	8 1 1 2 3 1 6 5
161	8 2 3 4 6 5 4 4
11	2 2
30	4 2 3 1
97	6 2 3 4 6 5

Note: In tables I and II the column headings are in the order X, A, X^{-1}, A^{-1} not the same order as in chapter one.

We would like to mention some of the difficulties encountered when writing the program. The first major difficulty is the complexity of the algorithm. It is considerably more involved than ordinary coset enumeration. A second difficulty is the problem of words and an associated problem of storage allocation. The words that are generated i.e. those in TABLE II are of arbitrary length. Hence a set amount of storage cannot be assumed for a particular word. We solved this problem by using one large array and storing the words in linear order. However this array has to be of quite large size initially but may not be used. Hence it is somewhat inefficient. Further the longer the words the more time it takes to manipulate them. Thus there is a considerable decrease in the speed of execution as compared to ordinary coset enumeration. One other difficulty was that of output. The data is handled in numeric form internally but must be translated into group terms. There is however no standardized set of symbols when working with groups hence the problem of translation was left to the user. A copy of the entire program is enclosed in a flap at the end of the thesis.

CHAPTER FIVE

In this chapter we would like to present several examples illustrating the usefulness of the new algorithm. There are four and the first is the following.

We would like to show that the group

$$G = \langle\ a,\ b,\ c\ ;\ a^{-1}ca = c^2,\ b^{-1}ab = a^2,\ c^{-1}bc = b^2\ \rangle$$

is the trivial group. The method we use is to pick a subgroup, to perform an extended coset enumeration and use the derived relations to prove G is the identity. So let H be the subgroup generated by

$$\{\ a,\ b^2\ \}$$

We define 3 cosets as follows

$$1 = H \qquad \bar{1} = e$$
$$2 = 1c^{-1} \qquad \bar{2} = c^{-1}$$
$$3 = 2c^{-1} \qquad \bar{3} = c^{-2}$$

From the definition of the subgroup we have immediately

$$1a = 1 \qquad \bar{1}a = a\bar{1}$$
$$1b^2 = 1 \qquad \bar{1}b^2 = b^2\bar{1}$$

Using this information and the definitions above, we fill in part of the relation tables

		c^{-1}	b	c	b^{-1}	b^{-1}
α	1	2	2	1		1
β	2	3	3	2	2	2

		a^{-1}	c	a	c^{-1}	c^{-1}
γ	3	2	1	1	2	3

The row beginning α gives us

$$2b = 2 \qquad \bar{2}\,b = b^2\bar{2}$$

We use the information obtained in row α to write row β and find

$$3b = 3 \qquad \bar{3}b = b^4\bar{3}$$

Now row γ gives us

$$3a^{-1} = 2 \qquad \bar{3}a^{-1} = a^{-1}\bar{2}$$

or $$2a = 3 \qquad \bar{2}a = a\bar{3}$$

The third relation is used to derive a redundancy

$$\begin{array}{cccccc} & b^{-1} & a & b & a^{-1} & a^{-1} \\ 2 & | \quad 2 & | \quad 3 & | \quad 3 & | & \underline{2 \; | \; 2} \end{array}$$

This row gives the new information (after rewriting the row)

$$2a^{-1} = 2 \qquad \bar{2}a^{-1} = ab^{-4}a^{-1}b^2\bar{2}$$

or $$2a = 2 \qquad \bar{2}a = b^{-2}ab^4a^{-1}\bar{2}$$

When we compare with the information obtained from row γ we see

$$2 = 3 \qquad a\bar{3} = b^{-2}ab^4a^{-1}\bar{2}$$

We substitute the known values of the coset representativesi.e.

$$\bar{2} = c^{-1}$$

$$\bar{3} = c^{-2}$$

and obtain the following identity

$$\begin{aligned} c^{-2} &= a^{-1}\, b^{-2}\, a\, b^4\, a^{-1}\, c^{-1} \\ c^{-1} &= a^{-1}\, b^{-2}\, a\, b^4\, a^{-1} \\ &= a^{-1}\, b^{-1}\, \underline{b^{-1}\, a\, b}\, b^3\, a^{-1} \\ &= a^{-1}\, b^{-1}\, a^2\, b^3\, a^{-1} \\ &= a^{-1}\, (b^{-1}\, a\, b)^2\, b^2\, a^{-1} \\ &= a^{-1}\, a^4\, b^2\, a^{-1} \\ &= a^3\, b^2\, a^{-1} \end{aligned}$$

So $$c = a\, b^{-2}\, a^{-3}.$$

We substitute this value of c in the original group relations to get the three new relations:

1. $b^{-1}ab = a^2$

2. $a^{-1}ca = c^2$

$$a^{-1}ab^{-2}a^{-3}a = ab^{-2}a^{-3}ab^{-2}a^{-3}$$

$$b^{-2}a^{-2} = ab^{-2}a^{-2}b^{-2}a^{-3}$$

$$b^{-2}a^{-2}a^{3}b^{2} = ab^{-2}a^{-2}$$

$$b^{-2}ab^{2} = ab^{-2}a^{-2}$$

$$b^{-1}\underline{b^{-1}a}bb = ab^{-2}a^{-2}$$

$$b^{-1}a^{2}b = ab^{-2}a^{-2}$$

$$a^{4} = ab^{-2}a^{-2}$$

$$a^{5} = b^{-2}$$

3. $c^{-1}bc^{-1} = b^2$

$$a^{3}b^{2}a^{-1}bab^{-2}a^{-3} = b^{2}$$

We simplify using relation 2.

$$a^{3}a^{-5}a^{-1}baa^{5}a^{-3} = a^{-5}$$

$$a^{-3}ba^{3} = a^{-5}$$

$$b = a^{-5}$$

We now use relation 3 and simplify relation 1.

1. $a^{5}aa^{-5} = a^{2}$

$$a = a^{2}$$

This of course implies

$$a = e.$$

Relation 3 immediately implies

$$b = e.$$

Since c has been defined in terms of a and b it is also true

$$c = e.$$

Thus the group G is the identity. The technique of coset enumeration allowed us to write c in terms of the subgroup generators a and b^2 and from that point on, straightforward algebraic manipulation gave us the proof.

Another example of some interest is the following. Let

$$G = \langle A, X ; X^{-1}A^2X = A^3 \rangle$$

Mendelsohn gives this as an example (Mendelsohn, 1967) to show the necessity of introducing Reidemeister-Schreier generators when attempting to find a presentation for a subgroup. If we let

$$H = \{ X, A^8 \}$$

then $H = G$ and further G. Higman has shown that in terms of X and A^8 G requires at least two defining relators. Now Mendelsohn states as there is only one coset, there is only one relation of the form

$$\mathcal{I}R\mathcal{I}^{-1} = e \qquad \mathcal{I} \text{ coset representative, } R \text{ relator}$$

His method now introduces Reidemeister-Schreier generators in order to get a second relation. His procedure is very roundabout which he admits in his paper. Our method is much simpler and more direct. We merely rewrite the subgroup generator A^8 to get the second relation. We will perform an extended coset enumeration on the subgroup

$$H_1 = \{ X, A^4 \}$$

The situation is very similar to the one above. Clearly $H_1 \supset H$ hence $H_1 = G$. Also H_1 requires at least two relators for its presentation. However the enumeration of cosets for H_1 is simpler than for H since only 4 cosets need be defined before collapse occurs while 8 need to be defined in the first case. Let

$$\alpha = X$$
$$\beta = A^4$$

Define

$$1 = H \qquad \bar{1} = e$$
$$2 = 1A \qquad \bar{2} = \bar{1}A = A$$
$$3 = 2A \qquad \bar{3} = \bar{2}A = A^2$$
$$4 = 3A \qquad \bar{4} = \bar{3}A = A^3$$

We use TABLE III and obtain the following information.

$$1X = 1 \qquad \bar{1}X = \alpha\bar{1}$$
$$4A = 1 \qquad \bar{4}A = \beta\bar{1}$$

We use the information in the relation and obtain

	X^{-1}	A	A	X	A^{-1}	A^{-1}	A^{-1}	
1		1	2	3	4	3	2	1

Thus

$$3X = 4$$
$$\bar{3}X = \bar{3}\, A^{-1}\, A^{-1}\, X\, A\, A\, A$$
$$= \alpha\bar{4}$$

At this point the coset multiplication table is as follows on the next page.

	A	A^{-1}	X	X^{-1}
$\bar{1}$	$\bar{2}$	$\beta^{-1}\bar{4}$	$\alpha\bar{1}$	$\alpha^{-1}\bar{1}$
$\bar{2}$	$\bar{3}$	$\bar{1}$		
$\bar{3}$	$\bar{4}$	$\bar{2}$	$\alpha\bar{4}$	
4	$\beta\bar{1}$	$\bar{3}$		$\alpha^{-1}\bar{3}$

We use the relation again and find

$$\begin{array}{ccccccc} X^{-1} & A & A & X & A^{-1} & A^{-1} & A^{-1} \end{array}$$

$$4 \mid \; 3 \mid 4 \mid 1 \mid 1 \mid \; 4 \mid \; \underline{3 \mid 4}$$

We find

$$3A^{-1} = 4$$

$$\bar{3}A^{-1} = \beta\alpha^{-1}\beta^{-1}\alpha\ \bar{4}$$

However from the above table we know

$$3A^{-1} = 2$$

$$\bar{3}A^{-1} = \bar{2}$$

Hence we may conclude

$$2 = 4$$

$$\bar{2} = \beta\alpha^{-1}\beta^{-1}\alpha\ \bar{4}$$

Multiplying row $\bar{4}$ in the above table by $\beta\alpha^{-1}\beta^{-1}\alpha$ gives:

1. $\bar{3} = \beta\alpha^{-1}\beta^{-1}\alpha\beta\ \bar{1}$
2. $\bar{1} = \beta\alpha^{-1}\beta^{-1}\alpha\ \bar{3}$

We also get the new information

$$\bar{2}X^{-1} = \beta\alpha^{-1}\beta^{-1}\ \bar{3}$$

We replace every occurrence of $\bar{4}$ by $\alpha^{-1}\beta\alpha\beta^{-1}\bar{2}$ and erase row $\bar{4}$. We now process redundancy #1. We multiply row $\bar{3}$ by $\beta^{-1}\alpha^{-1}\beta\alpha\beta^{-1}$ and we note the following:

3. $\bar{2} = \beta^{-1}\alpha^{-1}\beta\alpha\beta^{-1}\alpha^{-1}\beta\alpha\beta^{-1}\ \bar{2}$

4. $\beta^{-1}\alpha^{-1}\beta\alpha\beta^{-1}\ \bar{2} = \beta^{-1}\alpha^{-1}\beta\alpha\beta^{-1}\ \bar{2}$

5. $\alpha\ \bar{1} = \beta^{-1}\alpha^{-1}\beta\alpha^{2}\beta^{-1}\ \bar{2}$

We replace $\bar{3}$ by $\beta\alpha^{-1}\beta^{-1}\alpha\beta\bar{1}$ in the multiplication table and in redundancy #2. Redundancy #2 now gives the following

$$\bar{1} = \beta\alpha^{-1}\beta^{-1}\alpha\beta\alpha^{-1}\beta^{-1}\alpha\beta\ \bar{1}$$

or

$$\beta\alpha^{-1}\beta^{-1}\alpha\beta\alpha^{-1}\beta^{-1}\alpha\beta = e.$$

This is our first relation. Redundancy #3 gives the inverse of this relation. Redundancy #4 is trivial. To process redundancy #5, we multiply row $\bar{2}$ by $\alpha^{-1}\beta^{-1}\alpha^{-1}\beta\alpha^{2}\beta^{-1}$ and note:

6. $\bar{2} = \alpha^{-1}\beta^{-1}\alpha^{-1}\beta\alpha\beta^{-1}\alpha\beta\ \bar{1}$

7. $\beta^{-1}\alpha^{-1}\beta\alpha\beta^{-1}\ \bar{2} = \alpha^{-1}\beta^{-1}\alpha^{-1}\beta\alpha^{2}\beta^{-1}\ \bar{1}$

8. $\alpha^{-1}\ \bar{1} = \alpha^{-1}\ \bar{1}$

We now replace $\bar{2}$ by $\beta\alpha^{-2}\beta^{-1}\alpha\beta\alpha\bar{1}$ in the coset multiplication table which gives us

	A	A^{-1}	X	X^{-1}
$\bar{1}$	$\beta\alpha^{-2}\beta^{-1}\alpha\beta\alpha\bar{1}$	$\beta^{-1}\alpha^{-1}\beta\alpha^{-1}\beta^{-1}\alpha\beta\alpha\bar{1}$	$\alpha\bar{1}$	$\alpha^{-1}\bar{1}$

We also replace $\bar{2}$ in redundancies #6 and #7. We now process redundancy #6. The updated equation is

$$\beta\alpha^{-2}\beta^{-1}\alpha\beta\alpha\ \bar{1} = \alpha^{-1}\beta^{-1}\alpha^{-1}\beta\alpha\beta^{-1}\alpha\beta\ \bar{1}$$

or

$$\beta\alpha^{-2}\bar{\beta}^{-1}\alpha\beta\alpha\beta^{-1}\alpha^{-1}\beta\alpha^{-1}\beta^{-1}\alpha\beta\alpha = e.$$

This is our second relation. Redundancy #7 gives the inverse of this relation. Redundancy #8 is trivial. The extended coset enumeration is complete since all the tables are filled. Thus we can present the subgroup H_1 as follows:

$$H_1 = \langle \alpha, \beta ; I, II \rangle$$

where $\alpha = X$

$$\beta = A^4$$

$$I = \beta\alpha^{-1}\beta^{-1}\alpha\beta\alpha^{-1}\beta^{-1}\alpha\beta$$

$$II = \beta\alpha^{-2}\beta^{-1}\alpha\beta\alpha\beta^{-1}\alpha^{-1}\beta\alpha^{-1}\beta^{-1}\alpha\beta\alpha$$

Another point is shown by this example. If we attempt to use the Reidemeister-Schreier algorithm to get a presentation for H_1, the following results. There are two generators

$$S_{1,A}, S_{1,X}$$

and only one relation

$$S_{1,X}S_{1,A}^{2}S_{1,X}^{-1}S_{1,A}^{-3}$$

No new information is obtained. Even though $H_1 = G$ the Reidemeister-Schreier technique does not give a presentation for H_1 in terms of the generators of H_1. Thus in this instance, the extended coset enumeration is more fruitful, since we do obtain a presentation for H_1 in terms of the subgroup generators.

The third example is the group LF(2,p), that is the central quotient group of the special linear homogeneous group SL(n,q) of matrices of determinant 1 over the Galois Field of order p. The group is generated by the linear fractional transformations (mod p)

$$S: Z' = Z + 1$$

$$T: Z' = \frac{-1}{Z}$$

which, as matrices are

$$S = \begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix}$$

$$T = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}$$

In 1933 Frasch (Frasch, 1933, p.252) gave the following set of defining relations for the group. If we let

$$V = \begin{pmatrix} \alpha & C \\ 0 & \alpha^{-1} \end{pmatrix}$$

where α is a primitive root mod p then

$$(1)\quad LF(2,p) = \langle\, S,T,V \;;\; S^p = T^2 = V^{\frac{p-1}{2}} = (ST)^3 = (VT)^2 = e$$
$$V^{-1}SV = S^{\alpha^2} \,\rangle$$

with the extra relation

$$(TVS^{\alpha})^3 = e$$

when $p \equiv 1 \pmod 4$. We first prove that this is a set of defining relators and then we simplify the presentation somewhat. The proof of the first statement is adapted from a paper by Todd (Todd, 1932) and is simpler than Frasch's original proof. We show that the group contains three operations V, S, T and that they satisfy the given relations; conversely we show we can construct the group from the relations (1).

From the definition of S and T it is clear

$$S^p = T^2 = e.$$

Also by definition, the operation V is the following

$$V\colon Z' = \alpha^2 Z$$

Since α is a primitive root, V is an operation of period $\frac{1}{2}(p-1)$. The operation $V^{-1}SV$ can be described as follows

$$V^{-1}SV: Z = V^{-1}S: \alpha^2 Z$$

$$= V^{-1}: (\alpha^2 Z + \alpha^2)$$

$$= Z + \alpha^2$$

$$= S^{\alpha^2}: Z$$

Thus we can conclude

$$V^{-1}SV = S^{\alpha^2}.$$

Similarly

$$VTVT: Z = VTV: \frac{-1}{Z}$$

$$= VT: \frac{-1}{\alpha^2 Z}$$

$$= V: \frac{-Z}{\alpha^2}$$

$$= Z$$

Again we conclude

$$(VT)^2 = e.$$

The other relations in (1), namely

$$(ST)^3 = (TVS^{\alpha})^3 = e$$

are established in the same fashion. Thus the linear fractional group contains the three operations S, T, V and they satisfy (1). Thus the presentation (1) either defines the group LF(2,p) or has it as a factor group. To show that it is the group LF(2,p) we show that its order is the same as the order of LF(2,p) namely, $\frac{1}{2}p(p^2-1)$. In order to do this we perform a simple coset enumeration. Let H be the subgroup defined by

$$\{ S, V \} = \infty$$

Then, H subject to the relations in (1) is of order $\frac{1}{2}p(p-1)$ since all the elements in H can be expressed in the form

$$V^h S^k \qquad 0 \le h \le \frac{p-1}{2}, \qquad 0 \le k \le p-1$$

We now define p cosets $\mathit{0}, \mathit{1}, \mathit{2}, \ldots \mathit{p-1}$ in the following way. Let

$$\begin{aligned} \mathit{0} &= \infty \cdot T \\ \mathit{1} &= \mathit{0} \cdot S \\ \mathit{2} &= \mathit{1} \cdot S \\ &\ \ \vdots \\ \mathit{p-1} &= \mathit{p-2} \cdot S \end{aligned}$$

Then from the relator S^p we have

$$i \cdot S^j = i + j \pmod{p}$$

We next use the relation $(TV)^2 = e$

$$\begin{array}{ccccccccc} & T & & V & & T & & V & \\ \infty & & \underline{\mathit{0}} & & \underline{\mathit{0}} & & \infty & & \infty \end{array}$$

and we see that

$$\mathit{0} \cdot V = \mathit{0}$$

We now go to the relation $V^{-1}SV = S^{\alpha^2}$

$$\begin{array}{ccccccccc} & V^{-1} & & S & & V & & S^{-\alpha^2} & \\ \mathit{0} & & \mathit{0} & & \underline{\mathit{1}} & & \underline{\alpha^2} & & \mathit{0} \\ \alpha^2 & & \mathit{1} & & \underline{\mathit{2}} & & \underline{2\alpha^2} & & \alpha^2 \end{array}$$

Using the information $\mathit{0} \cdot V = \mathit{0}$ we then obtain $\mathit{1} \cdot V = \alpha^2$. This table enables us to find information of the form $j \cdot V$ assuming we know $(j-1) \cdot V$. Thus in general we have

$$i \cdot V = i\alpha^2.$$

To complete the coset enumeration we have to fill in the multiplication table for the operation T. First using the relator $(ST)^3$ we get

$$\begin{array}{ccccccc} & S & T & S & T & S & T \\ \infty & \infty & 0 & 1 & p-1 & 0 & \infty \end{array}$$

i.e. $\quad 1 \cdot T = p-1$

Now we use the relation $(TV)^2 = e$.

$$\begin{array}{ccccccccc} & T & & V & & T & & V & \\ 1 & & p-1 & & \underline{p-1\alpha^2} & & \underline{\alpha^{-2}} & & 1 \\ \alpha^{-2} & & p-1\alpha^2 & & \underline{p-1\alpha^4} & & \underline{\alpha^{-4}} & & \alpha^{-2} \\ & & & & \cdot & & & & \\ & & & & \cdot & & & & \\ & & & & \cdot & & & & \end{array}$$

This relation enables us to find all the information of the form

$$(p-1)\alpha^{2k} \cdot T = \alpha^{-2k} \qquad k = 1,2, \ldots$$

Now if $p \equiv -1 \pmod 4$, -1 is a quadratic nonresidue mod p, consequently each of the numbers $1,2, \ldots p-1$ is either a quadratic residue or the negative of a quadratic residue. Thus in this case the set

$$\{ \alpha^{-2k} \}$$

is a complete set of numbers x, $1 \le x \le p-1$ mod p. So the coset enumeration is finished since the multiplication table for the cosets is completed.

If $p \equiv 1 \pmod 4$ then -1 is a quadratic residue hence

$$\{ \alpha^{-2k} \}$$

is not a complete set of numbers. If this happens we use the extra relation

$$(S^{\alpha}TV)^3 = e$$

and gain the following information

		T	V	S^{α}	T	V	S^{α}	T	V	S^{α}
$\underline{\alpha}$	$\underline{-\alpha^{-1}}$	α	0	∞	∞	∞	0	0	α	

i.e. $\alpha \cdot T = -\alpha^{-1}$

We now use the relation $(TV)^2 = e$.

	T	V	T	V
$\underline{\alpha^3}$	$\underline{-\alpha^{-3}}$	$-\alpha^{-1}$	α	α^3
$\underline{\alpha^5}$	$\underline{-\alpha^{-5}}$	$-\alpha^{-3}$	α^3	α^5
		.		
		.		
		.		

We get the information

$$\alpha^{2k+1} \cdot T = -\alpha^{-2k-1} \qquad k = 0,1,2, \ldots$$

Thus when $p \equiv 1 (\mathrm{mod}\ 4)$ we use the extra relation and complete the coset enumeration. So in either case we have completed the coset enumeration and shown that the index of H in (1) is p+1. Since the order of H is $\frac{1}{2}p(p-1)$ we get that the order of the presentation (1) is $\frac{1}{2}p(p^2-1)$ proving it is a presentation of LF(2,p).

Frasch's presentation may be simplified in the following way. First we find an expression for V in terms of S and T.

$$\begin{aligned} e &= (S^{\alpha}TV)^3 \\ &= S^{\alpha}\, T\, \underline{V\, S^{\alpha}\, V^{-1}}\ \underline{V\, T\, V}\, S^{\alpha}\, T\, V \\ &= S^{\alpha}\, T\, S^{\alpha^{-1}}\, T\, S^{\alpha}\, T\, V \end{aligned}$$

or

$$V^{-1} = S^{\alpha}\, T\, S^{\alpha^{-1}}\, T\, S^{\alpha}\, T$$

Now we show $V^{\frac{p-1}{2}}$ is derivable from the other relations. First let us derive a few preliminary relations

$$(2)\quad V^jTV^jT = e$$

since

$$(TVT) = V^{-1}$$

$$(TVT)^j = V^{-j}$$

$$TV^jT = V^{-j}$$

$$(3)\quad V^{-j}SV^j = S^{\alpha^{2j}}$$

for

$$V^{-j}SV^j = V^{-j+1}S^{\alpha^2}V^{j-1}$$

$$\vdots$$

$$= S^{\alpha^{2j}}$$

Now

$$V^{2j} = V^j\ S\ T\ S\ T\ S\ T\ V^j$$

$$= \underline{V^j\ S\ V^{-j}}\ \underline{V^j\ T\ V^j}\ \underline{V^{-j}\ S\ V^j}\ \underline{V^{-j}\ T\ V^{-j}}\ \underline{V^j\ S\ V^{-j}}\ \underline{V^j\ T\ V^j}$$

$$= S^{\alpha^{-2j}}\ T\ S^{\alpha^{2j}}\ T\ S^{\alpha^{-2j}}\ T$$

If $p \equiv 1 (\text{mod } 4)$ let $j = \frac{p-1}{4}$, then

$$V^{\frac{p-1}{2}} = V^{2j}$$

$$= S^{\alpha^{-(\frac{p-1}{2})}}\ T\ S^{\alpha^{\frac{p-1}{2}}}\ T\ S^{\alpha^{-(\frac{p-1}{2})}}\ T$$

Now $\alpha^{-(\frac{p-1}{2})} \equiv -1 (\text{mod } p)$ and $\alpha^{\frac{p-1}{2}} \equiv -1 (\text{mod } p)$ since α is a primitive root, so

$$V^{\frac{p-1}{2}} = (S^{-1}T)^3$$

$$= e.$$

On the other hand if $p \equiv 3 \pmod 4$, let $j = \frac{p-3}{4}$ and then we have

$$V^{\frac{p-3}{2}} = S^{\alpha^{-(\frac{p-3}{2})}}\ T\ S^{\alpha^{\frac{p-3}{2}}}\ T\ S^{\alpha^{-(\frac{p-3}{2})}}\ T$$

Now $\alpha^{-(\frac{p-3}{2})} \equiv -\alpha$ and $\alpha^{\frac{p-3}{2}} \equiv -\alpha^{-1}$, so

$$\begin{aligned} V^{\frac{p-3}{2}} &= S^{-\alpha}\ T\ S^{-\alpha^{-1}}\ T\ S^{-\alpha}\ T \\ &= T(S^{\alpha}\ T\ S^{\alpha^{-1}}\ T\ S^{\alpha}\ T)^{-1}T \\ &= T(V^{-1})^{-1}T \qquad \text{from above} \\ &= T\ V\ T \\ &= V^{-1} \end{aligned}$$

i.e. $\quad V^{\frac{p-1}{2}} = e$

Thus we can delete the relator $V^{\frac{p-1}{2}}$ from Frasch's presentation. We now replace V in the two relations

$(VT)^2 = (S^{\alpha}TV)^3 = e$ to get

(4) $(VT)^2 = (V^{-1}T)^2 = (S^{\alpha}TS^{\alpha^{-1}}TS^{\alpha})^2 = e$

(5) $$\begin{aligned} (S^{\alpha}TV)^3 &= (V^{-1}TS^{-\alpha})^3 \\ &= (S^{\alpha}TS^{\alpha^{-1}}TS^{\alpha}TTS^{-\alpha})^3 \\ &= (S^{\alpha}TS^{\alpha^{-1}}T)^3 \end{aligned}$$

We now show that the relation $V^{-1}SV = S^{\alpha^2}$ is redundant.

$$\begin{aligned} V^{-1} &= S^{\alpha}TS^{\alpha^{-1}}TS^{\alpha}T \\ &= TS^{-\alpha^{-1}}TS^{-\alpha}TS^{-\alpha^{-1}}TS^{-\alpha}S^{\alpha}T \qquad \text{from (5)} \end{aligned}$$

So $\quad V = S^{\alpha^{-1}}TS^{\alpha}TS^{\alpha^{-1}}T$

We use (4)

$$(S^{\alpha}TS^{\alpha^{-1}}TS^{\alpha})^2 = e$$

$$= S^{\alpha}\ T\ S^{\alpha^{-1}}\ T\ S^{\alpha}\ S^{\alpha}\ T\ S^{\alpha^{-1}}\ T\ S^{\alpha}$$

$$= T\ S^{\alpha}\ T\ S^{\alpha^{-1}}\ T\ S^{\alpha}\ S^{\alpha}\ T\ S^{\alpha^{-1}}\ T\ S^{\alpha}\ T$$

$$S^{\alpha^{-1}} = S^{\alpha^{-1}}\ T\ S^{\alpha}\ T\ S^{\bar{\alpha}^{-1}}\ T\ S^{\alpha}\ S^{\alpha}\ T\ S^{\alpha^{-1}}\ T\ S^{\alpha}\ T$$

$$= V\ S^{\alpha}\ V^{-1}$$

Thus after deleting V, and $V^{-1}SV = S^{\alpha^2}$ we are left with

$$LF(2,p) = \langle\ S,\ T;\ S^p = T^2 = (ST)^3 = (S^{\alpha}TS^{\alpha^{-1}}T)^3 = (S^{\alpha}TS^{\alpha^{-1}}TS^{\alpha})^2 = e\ \rangle$$

A further simplification is possible if 2 is a primitive root. For we let $\alpha^{-1} = 2$ and $\alpha = \frac{p+1}{2}$ then the relation

$$(S^{\alpha}TS^{\alpha^{-1}}TS^{\alpha})^2 = e$$

is redundant. For it becomes

$$S^{\frac{p+1}{2}}\ T\ S^2\ T\ S^{\frac{p+1}{2}}\ S^{\frac{p+1}{2}}\ T\ S^2\ T\ S^{\frac{p+1}{2}}$$

$$= S^{\frac{p+1}{2}}\ T\ S^2\ T\ S\ T\ S^2\ T\ S^{\frac{p+1}{2}}$$

$$= S^{\frac{p+1}{2}}\ T\ S\quad S\ T\ S\ T\ S\quad S\ T\ S^{\frac{p+1}{2}}$$

$$= S^{\frac{p+1}{2}}\ \underline{T\ S\qquad T\qquad S\ T}\ S^{\frac{p+1}{2}}$$

$$= S^{\frac{p+1}{2}}\ S^{-1}\ S^{\frac{p+1}{2}}$$

$$= e$$

Thus when 2 is a primitive root

$$LF(2,p) = \langle\ S,\ T\ ;\ S^p = T^2 = (ST)^3 = (S^2TS^{\frac{p+1}{2}}T)^3 = e\ \rangle$$

Behr and Mennicke (1968) have proved that this presentation holds for all primes p. We will show that this last set of relations can be derived from the relations of Frasch. However the more difficult problem of deriving Frasch's relations from those of Behr and Mennicke has not been done in an algebraic manner. The proof that Behr and Mennicke gave was indirect. In order to derive the relations of Behr and Mennicke, it is of course only necessary to show

$$(S^2TS^{\frac{p+1}{2}}T)^3 = e.$$

To do this we show

$$(S^{\alpha^k}TS^{\alpha^{-k}}T)^3 = e \qquad k = 0,1,2,\ .\ .\ .$$

Suppose k is even i.e. $k = 2m$

$$\begin{aligned}(S^{\alpha^{2m}}TS^{\alpha^{-2m}}T)^3 &= S^{\alpha^{2m}}\ T\ S^{\alpha^{-2m}}\ T\ S^{\alpha^{2m}}\ T\ S^{\alpha^{-2m}}\ T\ S^{\alpha^{2m}}\ T\ S^{\alpha^{-2m}}\ T \\ &= V^{-m}SV^{m}TV^{m}SV^{-m}TV^{-m}SV^{m}TV^{m}SV^{-m}TV^{-m}SV^{m}TV^{m}SV^{-m}T \\ &= V^{-m}\ S\ T\ S\ T\ S\ T\ S\ T\ S\ T\ S\ T\ V^{m} \\ &= V^{-m}\ (ST)^6\ V^{m} \\ &= e.\end{aligned}$$

Suppose k is odd i.e. $k = 2m + 1$

$$\begin{aligned}&(S^{\alpha^{2m+1}}TS^{\alpha^{-2m-1}}T)^3 \\ &= S^{\alpha^{2m+1}}\ T\ S^{\alpha^{-2m-1}}\ T\ S^{\alpha^{2m+1}}T\ S^{\alpha^{-2m-1}}\ T\ S^{\alpha^{2m+1}}\ T\ S^{\alpha^{-2m-1}}\ T \\ &= V^{-m}S^{\alpha}V^{m}TV^{m}S^{\alpha^{-1}}V^{-m}TV^{-m}S^{\alpha}V^{m}TV^{m}S^{\alpha^{-1}}V^{-m}TV^{-m}S^{\alpha}V^{m}TV^{m}S^{\alpha^{-1}}V^{-m}T \\ &= V^{-m}(S^{\alpha}\ T\ S^{\alpha^{-1}}\ T\ S^{\alpha}\ T\ S^{\alpha^{-1}}\ T\ S^{\alpha}\ T\ S^{\alpha^{-1}}\ T)V^{m}\end{aligned}$$

$$= V^{-m}(S^{\alpha}\ T\ S^{\alpha^{-1}}\ T)^3 V^m$$
$$= e.$$

Now there exists k such that $\alpha^k \equiv 2 (\mathrm{mod}\ p)$ and also $\alpha^{-k} \equiv \frac{p+1}{2} (\mathrm{mod}\ p)$ since $\alpha^k \alpha^{-k} \equiv \alpha^0 \equiv 1 (\mathrm{mod}\ p)$. However we have proved for all k

$$(S^{\alpha^k} T S^{\alpha^{-k}} T)^3 = e$$

and so in particular

$$(S^2 T S^{\frac{p+1}{2}} T)^3 = e.$$

Thus we have proved that the relations of Behr and Mennicke can be derived from the relations of Frasch.

Our last example is a nontrivial coset enumeration. We consider the group

$$\langle\ A, B\ ;\ A^8 = B^7 = (AB)^2 = (A^{-1}B)^3 = e\ \rangle$$

which is denoted by (8, 7|2, 3) (Coxeter 1939). We wish to prove that the word

$$(A^2 B^4)^6$$

is the identity. Leech and Mennicke (1961) have already showed this but their proof had "surprising indirectness" (Leech, 1963, p.266). In order to prove this relation more directly we take the subgroup

$$H = \{\ A^2,\ A^{-1}B\ \}$$

Let
$$\alpha = A^2$$
$$\beta = A^{-1}B$$

and we note H is subject to the following relations

$$\alpha^4 = \beta^3 = (\alpha\beta)^2 = e.$$

However we know

$$H = \langle \alpha, \beta ; \alpha^4 = \beta^3 = (\alpha\beta)^2 = e \rangle$$

is a presentation for the symmetric group of order 24 (Magnus, Karrass & Solitar, 1966, p.21, #13). Thus after the coset enumeration is complete every element in (8, 7|2, 3) can be written as a word in H times a coset representative. Now H is a well known group hence (8, 7|2, 3) becomes well known in the sense that we can tell whether any given word in the group is the identity.

The order of (8, 7|2, 3) is known to be 10752 (Leech and Mennicke, 1961) and the order of H is 24. Thus the enumeration involved 448 distinct cosets. However there were many redundant cosets so that enumeration of over 1000 cosets was ultimately required before collapse occurred. The multiplication tables are too voluminous to reproduce here (approximately 18 pages would be required) but we do prove our original statement. We rewrite the word

$$(A^2B^4)^6$$

in terms of the generators of H and a coset representative. We find the coset representative is $\mathcal{T}$ and hence may be cancelled. We now simplify the resulting word using the relations valid in H and find it is the identity

$$\begin{aligned}
\mathcal{T}\,(A^2B^4)^6 &= \alpha\beta^{-1}\alpha^{-1}\alpha\beta^{-1}\alpha^{-1}\alpha^{-1}\beta\alpha\beta^2\beta^{-1}\alpha^{-1}\alpha\beta\cdot\mathcal{T} \\
&= \alpha\beta^{-2}\alpha^{-2}\beta\alpha\beta^2 \cdot e \\
&= \alpha\beta\alpha^2\beta\alpha\beta^2 \\
&= \alpha\beta\alpha\ \alpha\ \beta\alpha\beta\ \beta \\
&= \beta^{-1}\ \alpha\ \alpha^{-1}\ \beta \\
&= e.
\end{aligned}$$

BIBLIOGRAPHY

Behr, H. & Mennicke, J.:

1968 A Presentation of the Groups PSL(2,p).
Canad. J. Math., XX, 1432-1438.

Coxeter, H. S. M.:

1939 The Abstract Groups $G^{m,n,p}$.
Trans. Amer. Math. Soc., 45, 73-150.

Coxeter, H. S. M. & Moser, W. O. J.:

1964 Generators and Relations for Discrete Groups.
Springer, Berlin.

Frasch, H.:

1933 Die Erzeugenden der Hauptkongruenzgruppen für Primzahlstufen.
Math. Ann. 108, 229-252.

Hall, M.:

1959 The Theory of Groups.
Macmillan, New York.

Leech, J.:

1962 Some Definitions of Klein's Simple Group of Order 168 and Other Groups.
Proc. Glasgow Math. Assoc., 5, 166-175.

Leech, J.:

1963 Coset Enumeration on Digital Computers.
Proc. Camb. Phil. Soc., 59, 257-267.

Leech,J. & Mennicke, J.:

1961 Note on a Conjecture of Coxeter
Proc. Glasgow Math. Assoc., 5, 25-29.

Magnus, W., Karrass, A. & Solitar, D.:

1966 Combinatorial Group Theory
John Wiley, New York.

Mendelsohn, N.S.:

1964 An Algorithmic Solution for a Word Problem in Group Theory.
Canad. J. Math., 16, 509-516.

1965 Correction to: An Algorithmic Solution to a Word Problem in Group Theory.
Canad. J. Math., 17, 505.

1967 Defining Relations for Subgroups of Finite Index of Groups.
Computational Problems in Abstract Algebra.
Pergamon, Oxford, 43-44.

Mendelsohn, N. S. & Benson, C. T.:

1966 A Calculus for a Certain Class of Word Problems in Groups.
J. Comb. Th., 1, 202-208.

Moore, E. H.:

1897 Concerning the Abstract Groups of Order k! and ½k!.
Proc. London Math. Soc., (1), 28, 357-366.

Todd, J. A.:

1932 A Note on the Linear Fractional Group.
J. London Math. Soc., 7, 195-200.

Todd, J. A. & Coxeter, H. S. M.:

1936 A Practical Method for Enumerating Cosets of a Finite Abstract Group.
Proc. Edinburgh Math. Soc., (2), 5, 26-34.

Trotter, H. F.:

1964 A Machine Program for Coset Enumeration.
Canad. Math. Bull., 7, 357-368.

1966 A Program for the Coxeter-Todd Algorithm for the Enumeration of Cosets in Groups.
Math. Algorithms, 1, 12-18.

1967 Correction to: A Program for the Coxeter-Todd Algorithm for the Enumeration of Cosets in Groups.
Math. Algorithms, 2, 212.

```
      $WATBIG CAMPBELL,KP=29,PAGES=50,TIME=285,RUN=FREE
C     **
C     THIS PROGRAM DETERMINES THE INDEX OF A SUBGROUP IN A GROUP AND A
C     SET OF DEFINING RELATIONS FOR THE SUBGROUP
C     THIS PROGRAM ONLY HANDLES ONE GROUP PER RUN
C     **
C
      IMPLICIT INTEGER*2(A-Z)
C
C     EACH OF THE DIMENSIONS IN THE COMMON STATEMENT IS THE MAXIMUM OF
C     THE DIMENSIONS GIVEN BELOW
C     WHEN THE WORD "LENGTH" IS USED, EXPANDED LENGTH IS MEANT.
C
      COMMON N1,N6,N3,N7,NCI,N5,N2(3),N4(1),R(1,7),S(3,6),WTAB(1000,4)
     1,CTI3(1000,4),NREL(5,1000)
C
C     THE COMMON STATEMENT MUST BE ADJUSTED TO CONFORM TO THE INPUT DATA
C     N1-NUMBER OF SUBGROUP GENERATORS
C     N2(K)-NUMBER OF GROUP ELEMENTS IN SUBGROUP GENERATOR K
C     N3- NUMBER OF RELATORS
C     N4(K)-NUMBER OF GROUP ELEMENTS IN RELATOR K
C     N5-NUMBER OF GROUP GENERATORS
C     N6 - LENGTH OF LONGEST SUBGROUP GENERATOR
C     N7 - LENGTH OF LONGEST RELATOR
C     NCI-MAXIMUM NUMBER OF COSETS
C     R(I,J) - I IS THE NUMBER OF RELATORS AND J IS THE LENGTH OF THE I
C              TH RELATOR
C     S(I,J) - I IS THE NUMBER OF SUBGROUP GENERATORS AND J IS THE LENGTH
C              OF THE I TH SUBGROUP GENERATOR
C     WTAB(I,J) - VALUE OF (COSET I * GENERATOR J). IT WILL BE ANOTHER
C                 COSET
C     CTI3(I,J) - POINTER TO THE WORD THAT RESULTS WHEN COSET I
C                 MULTIPLIES GENERATOR J
C     NREL(I,J) - KEEPS TRACK OF ROWS WHICH ARE NOT COMPLETE. IF THE
C                 J TH ROW IN RELATOR I HAS VALUE 0 THEN THAT ROW IS
C                 COMPLETE OTHERWISE A NONZERO VALUE
C
C     THERE IS NO MAXIMUM FOR EACH OF THE ABOVE VARIABLES. HOWEVER THE
C     FOLLOWING FORMULA SHOULD BE USED TO DETERMINE THE AMOUNT OF CORE
C     NEEDED TO RUN THE PROGRAM.
C     CORE = 42,500 + 4000*N5 + 2*N3*N7 + N1*N6 + N1  + N3 + N7
C     THIS WILL RUN UP TO 1000 COSETS
C
      COMMON /AREA1/ CTAB1(20000),KEEP
C
C     CTAB1 - ARRAY WHICH KEEPS THE WORDS GENERATED IN LINEAR ORDER
C     KEEP - CURRENT AMOUNT OF NONEMPTY SPACE IN CTAB1
C
      COMMON /AREA2/ NCOSE(100),RDUN(100,2),RDUN1(100,2)
C
C     NCOSE - AN ARRAY TO KEEP TRACK OF COSETS WHICH ARE REDUNDANT BUT
C             ALREADY PROCESSED
C     RDUN(I,1) AND RDUN(I,2) ARE TWO COSETS WHICH ARE EQUAL I.E. WHEN
C     REDUNDANCIES OCCUR THE VALUES ARE PUT IN RDUN TO AWAIT PROCESSING
C     RDUN1 - POINTER TO THE WORDS ASSOCIATED WITH THE REDUNDANCY
C
      INTEGER*2 EOJ/1/,INDIC/1/,IBET/0/,NCOSET/1/,AMT/0/,AMT1/0/
C
C     EOJ - INDICATES WHEN THERE ARE NO BLANK SPACES IN THE RELATORS
C     INDIC - INDICATES WHEN THERE ARE NO BLANK SPACES IN THE SUBGROUP
```

```
C      IBET - NUMBER OF REDUNDANT COSETS AVAILABLE FOR REUSE
C      NCOSET - NUMBER OF COSETS CURRENTLY DEFINED
C      AMT - NUMBER OF REDUNDANCIES WAITING TO BE PROCESSED
C      AMT1 - NUMBER OF REDUNDANCIES WAITING TO BE PROCESSED
C
       N31=2*N5
       N3=N5
C
C      *
C      THIS IS THE INPUT SECTION
C
C      THE FIRST DATA CARD GIVES THE LENGTH OF EACH SUBGROUP GENERATOR
C      WHEN EXPANDED
C
       READ(5,310) (N2(J),J=1,N1)
C
C      THE SECOND INPUT CARD GIVES THE LENGTH OF EACH RELATOR EXPANDED
C
       READ(5,310) (N4(J),J=1,N3)
C
C      NEXT EACH SUBGROUP GENERATOR IS READ IN,ONE GENERATOR PER CARD.
C      THE INPUT DATA IS NUMERIC AS EACH GROUP GENERATOR IS GIVEN A NUMBER
C      AND ITS INVERSE IS GIVEN A NUMBER EQUAL TO THAT NUMBER PLUS THE
C      NUMBER OF GENERATORS. FOR EXAMPLE IF G IS GENERATED BY (A,B,C) WE
C      WOULD INPUT A AS 1,B AS 2,C AS 3,A INVERSE AS 4,B INVERSE  S 5,C
C      INVERSE AS 6. SO ACCBAB WOULD BE INPUTTED AS 1 3 3 2 1 2
C
       DO 202 I=1,N1
       K=N2(I)
       READ(5,310) (S(I,J),J=1,K)
   202 CONTINUE
C
C      FINALLY EACH RELATOR IS READ IN ONE TO A CARD IN THE SAME NUMERICAL
C      MANNER
C
       DO 203 I=1,N3
       K=N4(I)
       READ(5,310) (R(I,J),J=1,K)
   203 CONTINUE
   310 FORMAT(40I2)
C
C      *
C      *ECHO CHECK*
C
       WRITE(6,31) (N2(J),J=1,N1)
    31 FORMAT(1H1,1X,' THIS IS THE DATA INPUTTED '//1X' THE NUMBER OF ELE
      1MENTS IN THE SUB GROUP GENERATORS '/20X,40I2)
       WRITE(6,32) (N4(J),J=1,N3)
    32 FORMAT(1X,' THE NUMBER OF ELEMENTS IN THE RELATORS '/20X,40I2)
       WRITE(6,35)
    35 FORMAT(1X,' THE SUBGROUP GENERATORS ')
       DO 33 I=1,N1
       K=N2(I)
       WRITE(6,34) (S(I,J),J=1,K)
    33 CONTINUE
    34 FORMAT(20X,40I2)
       WRITE(6,36)
    36 FORMAT(1X,' THE RELATORS ')
       DO 37 I=1,N3
       K=N4(I)
```

```
      WRITE(6,33) (R(I,J),J=1,K)
   37 CONTINUE
   38 FORMAT(20X,40I2)
C
C      INITIALIZATION
C
      DO 200 I=1,NCI
      DO 200 J=1,N81
      CTIB(I,J)=0
      WTAB(I,J)=0
  200 CONTINUE
      DO 210 I=1,N3
      NREL(I,1)=1
      DO 210 J=2,NCI
      NREL(I,J)=0
  210 CONTINUE
C
C      START EXECUTION OF ALGORITHM
C
  240 CALL RUNTRJ(INDIC,NCOSET,EOJ,AMT,AMT1)
C
C      EOJ-INDICATOR OF WHEN TABLES ARE COMPLETE I.E. ALGORITHM TERMINATES
C      INDIC- INDICATOR THAT PROGRAM HAS FINISHED WITH SUBGROUP GENERATORS
C
      IF(EOJ.EQ.0.AND.INDIC.EQ.0) GO TO 230
C
C      AMT- THE NUMBER OF REDUNDANCIES TO BE PROCESSED
C
      IF(AMT.EQ.0) GO TO 260
      CALL ROUN2(AMT,AMT1,IBET)
      GO TO 240
  260 CALL DEFINE(INDIC,NCOSET,EOJ,IBET)
      IF(EOJ.EQ.0) GO TO 230
      GO TO 240
C
C      *
C      OUTPUT SECTION
C      FIRST OUTPUT IS THE NUMBER OF COSETS DEFINED AND THOSE WHICH ARE
C      REDUNDANT
C
  230 WRITE(6,270) NCOSET,(NCOSE(J),J=1,IBET)
  270 FORMAT(///1X,' NUMBER OF COSETS DEFINED '5X,I5//1X,' REDUNDANT COS
     1ETS '(16X20I4))
C
C      NEXT OUTPUT IS THE INDEX OF THE SUBGROUP IN THE  GROUP
C
      JOK = NCOSET-IBET
      WRITE(6,271) JOK
  271 FORMAT(//1X,' THE INDEX OF THE SUBGROUP IS '//12X14//)
C
C      THE NEXT OUTPUT IS TABLE I
C
      WRITE(6,218)
      DO 217 I=1,NCOSET
      WRITE(6,33)  I,(WTAB(I,J),J=1,N81)
   33 FORMAT(1X,I3,':',20I5)
  217 CONTINUE
      WRITE(6,40)
   40 FORMAT(//)
C
```

```
C     NEXT OUTPUT IS TABLE 2-WHICH CONTAINS LOCATION NUMBERS OF THE
C     ACTUAL ENTRIES WHICH ARE PRINTED BELOW
C
      WRITE(6,201)
      DO 215 I=1,NCOSET
      WRITE(6,200) I,(CTIB(I,J),J=1,N81)
  215 CONTINUE
      WRITE(6,40)
C
C     THE LAST OUTPUT IS THE ENTRIES TO TABLE 2 PRINTED SEQUENTIALLY
C     THE NUMBERS PRINTED REFER TO SUBGROUP GENERATORS AND THEIR INVERSES
C
      WRITE(6,263)
  263 FORMAT(1X,' THE ENTRIES TO TABLE II WHERE THE FIRST NUMBER IS THE
     1LOCATION NUMBER IN THE TABLE ABOVE,'/1X,' THE SECOND NUMBER IS ONE
     1 MORE THAN THE NUMBER OF SYMBOLS IN THE WORD AND THEN THE WORD IS
     1'/'  PRINTED BELOW IN NUMERIC FORM. EACH NUMBER REPRESENTS A SUBGR
     1OUP GENERATOR ')
      DO 255 I=1,NCOSET
      DO 255 J=1,N81
      KK=CTIB(I,J)
      IF(KK.EQ.0) GO TO 255
      MN=CTAB1(KK)-1+KK
      WRITE(6,259) CTIB(I,J),(CTAB1(P),P=KK,MN)
  255 CONTINUE
  259 FORMAT(/1X,I6,6X,I6/(17X40I2))
  218 FORMAT(1X' WTAB-MULTIPLACATION TABLE FOR COSETS')
  201 FORMAT(1X' CTIB-POINTERS TO LOCATION OF COSET REPRESENTATIVES')
  200 FORMAT(1X,I3':',20I5)
C     *
C
      STOP
      END


      BLOCK DATA
      IMPLICIT INTEGER*2(A-Z)
      COMMON N1,N6,N3,N7,NCI,N5,N2(3),N4(1),R(1,7),S(3,6),WTAB(1000,4)
     1,CTIB(1000,4),NREL(5,1000)
      COMMON /AREA1/ CTAB1(20000),KEEP
C
C     DATA STATEMENT MUST BE ADJUSTED TO CONFORM TO INPUT DATA
C     N1,N3,N5,N6,N7 ARE PUNCHED ACCORDING TO THEIR VALUE FOR A PARTCU-
C     LAR GROUP
C
      DATA KEEP/1/,N3/1/,N5/2/,N1/3/,N6/6/,N7/6/,NCI/1000/
      END


      INTEGER FUNCTION CALC*2(IJ,INTER,K,SW)
C
C     THIS SUBROUTINE SOLVES  FOR NEW INFORMATION
C
      IMPLICIT INTEGER*2(A-Z)
      COMMON N1,N6,N3,N7,NCI,N5,N2(3),N4(1),R(1,7),S(3,6),WTAB(1000,4)
     1,CTIB(1000,4),NREL(5,1000)
      COMMON /AREA1/ CTAB1(20000),KEEP
C
C     RS MUST BE DIMENSIONED AS THE NUMBER OF RELATORS BY THE LENGTH OF
C     THE LONGEST RELATOR
```

```
C       INTER MUST BE DIMENSIONED AS LONG AS THE LONGEST RELATOR
C       THESE ARRAYS ARE TEMPERARY STORAGE ARRAYS
C
        DIMENSION INTER(50),RS(5,48)
        NB1=2*N5
        NB=N5
C
C       SW- TELLS WHETHER INFORMATION OCCURS IN SUBGROUP GENERATOR OR RELATOR
C
        IF(SW.EQ.2) GO TO 700
C
C       SUBGROUP GENERATOR
C
        SW1=N2(K)
        DO 712 I=1,N1
        KLK=N2(I)
        DO 712 J=1,KLK
        RS(I,J)=S(I,J)
    712 CONTINUE
        GO TO 704
C
C       RELATOR
C
    700 SW1=N4(K)
        DO 710 I=1,N3
        KLK=N4(I)
        DO 710 J=1,KLK
        RS(I,J)=R(I,J)
    710 CONTINUE
    704 LL=0
C
C
C       IF INFORMATION OCCURS AT FIRST SYMBOL PROGRAM SKIPS TO 734
        IF(IJ.EQ.1) GO TO 734
        IJ1=IJ+2
        IJ2=IJ+1
C
C       SOLVES  BACKWARDS FROM POINT OF INFORMATION TO FIRST SYMBOL
C
        DO 732 N=2,IJ
        ITT=INTER(IJ1-N)
        IIT=RS(K,IJ2-N)+NB
        IIT=ODD(IIT,NB1)
        WORD=CTIB(ITT,IIT)
        IF(WORD.EQ.0) GO TO 732
        KK=CTAB1(WORD)-1
        KEEP1=KEEP+LL
        DO 735 MN=1,KK
        CTAB1(KEEP1+MN)=CTAB1(WORD+MN)
    735 CONTINUE
        LL=LL+KK
    732 CONTINUE
C
C       IF INFORMATION OCCURS IN A RELATOR PROGRAM SKIPS NEXT TWO LINES
C       OTHERWISE THE NUMBER OF THE SUBGROUP GENERATOR IS ADDED TO THE
C       INFORMATION SO FAR ATTAINED
C
    734 IF(SW.EQ.2) GO TO 727
        CTAB1(KEEP+LL+1)=K
        LL=LL+1
```

```
C
C     IF INFORMATION OCCURS AT LAST SYMBOL PROGRAM SKIPS TO 737
C
  727 IF(IJ.EQ.SW1) GO TO 737
      SW12=SW1+1
      II=IJ+2
      SW123 =SW12+IJ+2
      SW124=SW123-1
C
C     PROGRAM NOW SOLVES BACKWARDS FROM THE LAST SYMBOL TO THE POINT OF
C     INFORMATION AND COMPLETES THE SOLUTION
C
      DO 746 N=II,SW12
      ITT=INTER(SW123-N)
      IIT=RS(K,SW124-N)+N8
      IIT=DOD(IIT,NB1)
      WORD=CTIB(ITT,IIT)
      IF(WORD.EQ.0) GO TO 746
      KK=CTAB1(WORD)-1
      KEEP1=KEEP+LL
      DO 747 MN=1,KK
      CTAB1(KEEP1+MN)=CTAB1(WORD+MN)
  747 CONTINUE
      LL=LL+KK
  746 CONTINUE
  737 IF(LL.NE.0) GO TO 750
      CALC=0
      RETURN
  750 CTAB1(KEEP)=LL+1
      CALC=CHECK(KEEP)
      KEEP=KEEP+CTAB1(KEEP)
      RETURN
      END


      SUBROUTINE RDUN2(AMT,AMT1,IBET)
C
C     THIS SUBROUTINE DEALS WITH REDUNDANCIES
C
      IMPLICIT INTEGER*2(A-Z)
      COMMON N1,N6,N3,N7,NCI,N5,N2(3),N4(1),R(1,7),S(3,6),WTAB(1000,4)
     1,CTIB(1000,4),NREL(5,1000)
      COMMON /AREA1/ CTAB1(20000),KEEP
      COMMON /AREA2/ NCOSE(100),RDUN(100,2),RDUN1(100,2)
      NB1=2*N5
   33 N8=N5
C
C     THIS SECTION DECIDES WHICH IS THE SMALLER OF THE TWO COSETS AND
C     NAMES IT NMIN AND NAMES THE LARGER NMAX AND FORMS THE WORD SPEC
C     SPEC- THE** WORD ** SUCH THAT  NMIN=WORD*NMAX
C
  304 IF(RDUN(AMT,1)-RDUN(AMT,2)) 107,108,109
  109 NMIN=RDUN(AMT,2)
      NMAX=RDUN(AMT,1)
      IT=RDUN1(AMT1,2)
      ITT=RDUN1(AMT1,1)
      SPEC=NSUM(IT,ITT)
      GO TO 75
  107 NMIN=RDUN(AMT,1)
      NMAX=RDUN(AMT,2)
```

```
      IT=RDUN1(AMT1,1)
      ITT=RDUN1(AMT1,2)
      SPEC=NSUM(ITT,IT)
      SPEC=INV(SPEC)
   75 IBET=IBET+1
C
C     NMAX IS FILED AS A REUSABLE COSET
C
      NCOSE(IBET)=NMAX
C
C     THE PROGRAM ZEROES OUT THE NMAX ROW IN RELATION TABLE
C
      DO 88 I=1,N3
      NREL(I,NMAX)=0
   88 CONTINUE
      RDUN(AMT,2)=0
C
C     ZERO OUT REDUNDANCIES DEALT WITH AND DECREASE NUMBER OF REDUNDANCIES
C     BY ONE
C
      RDUN(AMT,1)=0
      RDUN1(AMT1,1)=0
      RDUN1(AMT1,2)=0
      AMT=AMT-1
      AMT1=AMT1-1
C
C     COMPARE ROWS NMIN AND NMAX IN TABLE 1
C     IF BOTH ARE NONEMPTY WE HAVE A FURTHER REDUNDANCY
C     IF NMAX IS NONEMPTY AND NMIN IS EMPTY THE INFORMATION IS TRANSFERRED
C     IF NMAX IS EMPTY THEN PROGRAM SKIPS TO NEXT COLUMN
C
      DO 100 I=1,N81
      IF(WTAB(NMAX,I).EQ.0) GO TO 100
      IF(WTAB(NMIN,I).EQ.0) GO TO 103
      AMT=AMT+1
      AMT1=AMT1+1
      RDUN(AMT,1)=WTAB(NMIN,I)
      RDUN(AMT,2)=WTAB(NMAX,I)
      IT=CTIB(NMIN,I)
      RDUN1(AMT1,1)=IT
      ITT=CTIB(NMAX,I)
      ITT=NSUM(SPEC,ITT)
      RDUN1(AMT1,2)=INV(ITT)
  317 GO TO 100
  103 WTAB(NMIN,I)=WTAB(NMAX,I)
      IT=CTIB(NMAX,I)
      CTIB(NMIN,I)=NSUM(SPEC,IT)
  100 CONTINUE
C
C     WE REPLACE ALL OCCURRENCES OF NMAX BY NMIN IN TABLES 1 AND 2
C     THE ROW LABELLED NMAX IS ZEROED OUT IN TABLES 1 AND 2
C
      SPEC1=INV(SPEC)
      DO 105 I=1,N81
      RESULT=WTAB(NMAX,I)
      WTAB(NMAX,I)=0
      IF(RESULT.EQ.0) GO TO 105
      IT=I+N8
      IT=DUD(IT,N81)
      ITT=CTIB(NMAX,I)
```

```
      CTIB(NMAX,I)=0
      IF(WTAB(RESULT,IT).EQ.NMIN) GO TO 105
      WTAB(RESULT,IT)=NMIN
      ITT=INV(ITT)
      CTIB(RESULT,IT)=NSUM(ITT,SPEC1)
  105 CONTINUE
      IF(AMT.EQ.0) RETURN
C
C     THE LIST OF REMAINING REDUNDANCIES IS CHECKED AND OCCURRENCES OF
C     NMAX ARE REPLACED BY NMIN
C
      DO 400 I=1,AMT
      DO 400 J=1,2
      IF(RDUN(I,J).NE. NMAX) GO TO 400
  404 RDUN(I,J)=NMIN
      IF(J.EQ.2) GO TO 405
      ITT=SPEC1
      IT=RDUN1(I,J)
      RDUN1(I,J)=NSUM(IT,ITT)
      GO TO 400
  405 RDUN1(I,J)=NSUM(SPEC,RDUN1(I,J))
  400 CONTINUE
      IF(AMT.NE. 0) GO TO 98
      RETURN
C
C     THIS SECTION DEALS WITH THE CASE WHERE NMIN=NMAX. IF A RELATION OCCURS
C     IT IS PRINTED
C
  108 RDUN(AMT,1)=0
      RDUN(AMT,2)=0
      AMT=AMT-1
      SYSGEN=NSUM(RDUN1(AMT1,2),RDUN1(AMT1,1))
      IF(SYSGEN.EQ.0) GO TO 110
      LIT=SYSGEN+CTAB1(SYSGEN)-1
      WRITE(6,119) (CTAB1(I),I=SYSGEN,LIT)
  119 FORMAT(//1X,' A RELATOR DERIVED FROM REDUNDANCIES. THE FIRST NUMBE
     1R IS ONE MORE THAN THE LENGTH OF THE RELATOR.'/1X,' THE REMAINING
     1NUMBERS ARE THE RELATOR ITSELF '/(1XI4,5X,40I2))
  110 RDUN1(AMT1,1)=0
      RDUN1(AMT1,2)=0
      AMT1=AMT1-1
C
C     IF THERE ARE FURTHER REDUNDANCIES THEY ARE PROCESSED OTHERWISE
C     RETURN TO M/PROG
C
      IF(AMT.NE.0) GO TO 98
      RETURN
      END


      SUBROUTINE RUNTRJ(INDIC,NCOSET,EQU,AMT,AMT1)
C
C     THIS SUBROUTINE GOES THROUGH THE SUBGROUP GENERATORS AND RELATORS
C     TO TRY AND PICK UP NEW INFORMATION
C
      IMPLICIT INTEGER*2(A-Z)
      COMMON N1,N6,N3,N7,NC1,N5,N2(3),N4(1),R(1,7),S(3,6),WTAB(1000,4)
     1,CTIB(1000,4),NREL(5,1000)
      COMMON /AREA1/ CTAB1(20000),KEEP
      COMMON /AREA2/ NCOSE(100),RDUN(100,2),RDUN1(100,2)
```

```
      DIMENSION INTER(50)
      N81=2*N5
      N8=N5
C
C     PROGRAM CHECKS TO SEE IF THE SUBGROUP GENERATORS HAVE ALL BEEN
C     REWRITTEN I.E. NO BLANK SPACES OCCUR
C
      IF(INDIC.EQ.0) GO TO 16
C
C     THE FIRST SECTION DEALS WITH SUBGROUP GENERATORS
C
      DO 10 K=1,N1
C
C     WHENEVER A ROW IS COMPLETED IT IS ZEROED BECAUSE NO FURTHER
C     INFORMATION CAN BE GAINED
C
      IF(S(K,1).EQ.0) GO TO 10
      N2A=N2(K)
      N2K=N2(K)+1
      INTER(1)=1
C
C     THE PROGRAM GOES FORWARD IN A ROW AS FAR AS POSSIBLE
C
      DO 12 I=1,N2A
      IIT=S(K,I)
      ITT=INTER(I)
      INTER(I+1)=WTAB(ITT,IIT)
      IF(INTER(I+1).NE. 0) GO TO 12
      INTER(N2K)=1
      IF(I.EQ.N2A) GO TO 26
      I1=N2(K)
      II=I+1
C
C     AFTER GOING FORWARD IT GOES BACKWARD REMEMBERING WHERE IT STOPPED
C     IN THE FORWARD DIRECTION
C
      DO 14 J=II,I1
      ITI=INTER(N2K+II-J)
      TIT=S(K,N2K+I-J)+N8
      TIT=MOD(TIT,N81)
      INTER(N2K+I-J)=WTAB(ITI,TIT)
      IF(INTER(N2K+I-J).EQ.0) GO TO 10
   14 CONTINUE
C
C     THE THREE CASES ARE DEALT WITH
C                    -NO INFORMATION GO TO 10 I.E. TRY ANOTHER GENERATOR
C                    -NEW INFORMATION GO TO 26
C                    -REDUNDANCY GO TO 32 OR 30 DEPENDING ON THE PLACE OF
C                     OCCURRENCE
C
      GO TO 26
   12 CONTINUE
C
C     IF THE ENTIRE ROW IS PROCESSED FORWARDS EITHER A REDUNDANCY SO
C     PROGRAM SKIPS TO 30 OR NO NEW INFORMATION SO THE ROW IS ZEROED OUT
C
      IF(INTER(N2K).NE.1) GO TO 30
      S(K,1)=0
      GO TO 10
C
```

```
      DIMENSION INTER(50)
      NB1=2*N5
      NB=N5
C
C     PROGRAM CHECKS TO SEE IF THE SUBGROUP GENERATORS HAVE ALL BEEN
C     REWRITTEN I.E. NO BLANK SPACES OCCUR
C
      IF(INDIC.EQ.0) GO TO 16
C
C     THE FIRST SECTION DEALS WITH SUBGROUP GENERATORS
C
      DO 10 K=1,N1
C
C     WHENEVER A ROW IS COMPLETED IT IS ZEROED BECAUSE NO FURTHER
C     INFORMATION CAN BE GAINED
C
      IF(S(K,1).EQ.0) GO TO 10
      N2A=N2(K)
      N2K=N2(K)+1
      INTER(1)=1
C
C     THE PROGRAM GOES FORWARD IN A ROW AS FAR AS POSSIBLE
C
      DO 12 I=1,N2A
      IIT=S(K,I)
      ITT=INTER(I)
      INTER(I+1)=WTAB(ITT,IIT)
      IF(INTER(I+1).NE. 0) GO TO 12
      INTER(N2K)=1
      IF(I.EQ.N2A) GO TO 26
      I1=N2(K)
      II=I+1
C
C     AFTER GOING FORWARD IT GOES BACKWARD REMEMBERING WHERE IT STOPPED
C     IN THE FORWARD DIRECTION
C
      DO 14 J=II,I1
      ITI=INTER(N2K+II-J)
      TIT=S(K,N2K+I-J)+NB
      TIT=ODD(TIT,NB1)
      INTER(N2K+I-J)=WTAB(ITI,TIT)
      IF(INTER(N2K+I-J).EQ.0) GO TO 10
   14 CONTINUE
C
C     THE THREE CASES ARE DEALT WITH
C                    -NO INFORMATION GO TO 10 I.E. TRY ANOTHER GENERATOR
C                    -NEW INFORMATION GO TO 26
C                    -REDUNDANCY GO TO 32 OR 30 DEPENDING ON THE PLACE OF
C                     OCCURRENCE
C
      GO TO 26
   12 CONTINUE
C
C     IF THE ENTIRE ROW IS PROCESSED FORWARDS EITHER A REDUNDANCY SO
C     PROGRAM SKIPS TO 30 OR NO NEW INFORMATION SO THE ROW IS ZEROED OUT
C
      IF(INTER(N2K).NE.1) GO TO 30
      S(K,1)=0
      GO TO 10
C
```

```
C     THIS SECTION DEALS WITH NEW INFORMATION
C
   25 ITIT=INTER(I+1)
      ITTT=S(K,I)+NB
      ITTT=MOD(ITTT,NB1)
      IF(WTAB(ITIT,ITTT).NE. 0) GO TO 32
      WTAB(ITIT,ITTT)=INTER(I)
      WTAB(ITT,IIT)=INTER(I+1)
      SW=1
      CTIB(ITT,IIT)=CALC(I,INTER,K,SW)
      KJ=CTIB(ITT,IIT)
      CTIB(ITIT,ITTT)=INV(KJ)
      GO TO 10
C
C     THIS SECTION DEALS WITH REDUNDANCIES THAT OCCUR ANYWHERE BUT AT
C     THE LAST SYMBOL
C
   32 AMT=AMT+1
      RDUN(AMT,1)=WTAB(ITIT,ITTT)
      ITT=INTER(I-1)
      IIT=S(K,I-1)
      RDUN(AMT,2)=WTAB(ITT,IIT)
      AMT1=AMT1+1
      RDUN1(AMT1,1)=CTIB(ITIT,ITTT)
      SW=1
      RDUN1(AMT1,2)=CALC(I,INTER,K,SW)
      GO TO 10
C
C     THIS SECTION DEALS WITH REDUNDANCIES THAT OCCUR AT THE LAST SYMBOL
C
   30 AMT=AMT+1
      ITT=INTER(N2A)
      ITIT=S(K,N2A)
      RDUN(AMT,1)=WTAB(ITT,ITIT)
      RDUN(AMT,2)=1
      AMT1=AMT1+1
      RDUN1(AMT1,1)=CTIB(ITT,ITIT)
      IM=N2A
      SW=1
      RDUN1(AMT1,2)=CALC(IM,INTER,K,SW)
   10 CONTINUE
   16 DO 18 K=1,N3
C
C     THIS SECTION PARALLELS THE FIRST EXCEPT IT DEALS WITH THE RELATION
C     TABLE
C
      N4K=N4(K)+1
C
C     NCOSET IS THE CURRENT NUMBER OF COSETS DEFINED
C
      DO 20 J=1,NCOSET
      INTER(1)=NREL(K,J)
C
C     IF A ROW IS COMPLETE IT IS ZEROED OUT BECAUSE NO NEW INFORMATION
C     CAN OCCUR THERE
C
      IF(INTER(1).EQ.0) GO TO 20
      N4A=N4(K)
C
C     EACH ROW IN TABLE 4 IS PROCESSED FROM THE RIGHT AS FAR AS POSSIBLE.
```

```
C
      DO 22 I=1,N4A
      IIT=R(K,I)
      ITI=INTER(I)
      INTER(I+1)=WTAB(ITI,IIT)
      IF(INTER(I+1).EQ.0) GO TO 23
   22 CONTINUE
C
C     IF THE ENTIRE ROW IS PROCESSED EITHER A REDUNDANCY OR NO NEW
C     INFORMATION SO THE ROW IS ZEROED OUT
C
      IF(INTER(N4K).NE.NREL(K,J)) GO TO 71
      NREL(K,J)=0
      GO TO 20
   23 INTER(N4K)=NREL(K,J)
      IF(I.EQ.N4(K)) GO TO 28
      II=I+1
      N4K1=N4K+II
      N4K2=N4K+I
C
C     THE ROW IS NOW PROCESSED FROM THE LEFT AS FAR AS POSSIBLE AND 3 CASES
C     ARISE
C                   -INFORMATION OCCURS GO TO 28
C                   -REDUNDANCY OCCURS GO TO 52 OR 71 DEPENDING ON THE LOCATION
C                    OF THE BREAK
C                   -THERE ARE STILL BLANK SPACES SO ANEW ROW IS CONSIDERED
C
      DO 24 L=II,N4A
      ITT=INTER(N4K1-L)
      TIT=R(K,N4K2-L)+NB
      TIT=DDD(TIT,NB1)
      INTER(N4K2-L)=WTAB(ITT,TIT)
      IF(INTER(N4K2-L).EQ.0) GO TO 20
   24 CONTINUE
C
C     THIS SECTION DEALS WITH NEW INFORMATION
C
   28 ITIT=INTER(I+1)
      ITTT=R(K,I)+NB
      ITTT=DDD(ITTT,NB1)
      IF(WTAB(ITIT,ITTT).NE. 0) GO TO 52
      WTAB(ITIT,ITTT)=INTER(I)
      WTAB(ITI,IIT)=INTER(I+1)
      SW=2
      CTIB(ITI,IIT)=CALC(I,INTER,K,SW)
      KJ=CTIB(ITI,IIT)
      CTIB(ITIT,ITTT)=INV(KJ)
      NREL(K,J)=0
      GO TO 20
C
C     THIS SECTION DEALS WITH REDUNDANCIES THAT OCCUR ANYWHERE BUT AT
C     THE LAST SYMBOL
C
   52 AMT=AMT+1
      ROUN(AMT,1)=WTAB(ITIT,ITTT)
      ROUN(AMT,2)=INTER(I)
      AMT1=AMT1+1
      ROUN1(AMT1,1)=CTIB(ITIT,ITTT)
      SW=2
      ROUN1(AMT1,2)=CALC(I,INTER,K,SW)
```

```
      NREL(K,J)=0
      GO TO 20
C
C     THIS SECTION DEALS WITH REDUNDANCIES THAT OCCUR AT THE LAST SYMBOL
C
   71 AMT=AMT+1
      ITT=INTER(N4A)
      ITIT=R(K, N4A)
      RDUN(AMT,1)=NREL(K,J)
      RDUN(AMT,2)=WTAB(ITT,ITIT)
      AMT1=AMT1+1
      IT=R(K,2)+N8
      IT=DOD(IT,N81)
      RDUN1(AMT1,1)=CTIB(INTER(2),IT)
      SW=2
      IM=N4(K)
      RDUN1(AMT1,2)=CALC(IM,INTER,K,SW)
      ITI=R(K,N4A)+N8
      ITI=DOD(ITI,N81)
      ITT=INTER(N4K)
      ITT=CTIB(ITT,ITI)
      RDUN1(AMT1,2)=NSUM(RDUN1(AMT1,2),ITT)
      IF=INV(RDUN1(AMT1,1))
      IL=RDUN1(AMT1,2)
      RDUN1(AMT1,2)=NSUM(IL,IF)
      NREL(K,J)=0
   20 CONTINUE
   13 CONTINUE
      RETURN
      END


      SUBROUTINE DEFINE(INDIC,NCOSET,EOJ,IBET)
C
C     THIS SUBROUTINE DEFINES NEW COSETS WHEN THE TABLES ARE NOT COMPLETE
C     AND THERE ARE NO REDUNDANCIES TO BE PROCESSED
C     THERE ARE TWO SECTIONS,ONE FOR DEFINING COSETS IN A SUBGROUP
C     GENERATOR AND A SECTION FROM STATEMENT 15 WHERE COSETS ARE DEFINED
C     FROM THE BLANKS IN THE TABLE OF RELATORS
C
      IMPLICIT INTEGER*2(A-Z)
      COMMON N1,N6,N3,N7,NCI,N5,N2(3),N4(1),R(1,7),S(3,6),WTAB(1000,4)
     1,CTIB(1000,4),NREL(5,1000)
      COMMON /AREA2/ NCOSE(100),RDUN(100,2),RDUN1(100,2)
      N81=2*N5
      N8=N5
      IF(INDIC.EQ.0) GO TO 15
C
C     THE PROGRAM CHECKS THROUGH ROWS IN THE SUBGROUP GENERATORS UNTIL
C     THE FIRST BLANK SPACE IS FOUND AND THEN A NEW COSET IS DEFINED
C
      DO 3 K=1,N1
      INTERM=1
      N2ALT=N2(K)-1
      IF(N2ALT.EQ.0) GO TO 3
      IF(S(K,1).EQ.0) GO TO 3
      DO 5 L=1,N2ALT
      IIT=S(K,L)
      INTERI=WTAB(INTERM,IIT)
      IF(INTERI.NE. 0) GO TO 6
```

```
      WTAB(INTERM,IIT)=NCOSET+1
      CTIB(INTERM,IIT)=0
      IITT=IIT+N5
      IITT=ODD(IITT,NB1)
C
C     THE NUMBER OF COSETS DEFINED IS INCREASED BY ONE
C
      NCOSET=NCOSET+1
      WTAB(NCOSET,IITT)=INTERM
      CTIB(NCOSET,IITT)=0
C
C     ROWS ARE SET UP IN TABLE 4 FOR THE NEW COSET
C
      DO 19 J=1,N3
      NREL(J,NCOSET)=NCOSET
   19 CONTINUE
      RETURN
    6 INTERM=INTERI
    5 CONTINUE
    3 CONTINUE
C
C     IF THERE ARE NO BLANKS IN THE SUBGROUP GENERATORS THEN THE VALUE
C     OF INDIC IS CHANGED
C
      INDIC=0
   15 YU4=0
C
C     THIS SECTION PARALLELS THE FIRST AS THE ROWS IN THE RELATION TABLE
C     ARE SEARCHED FOR THE FIRST BLANK SPOT
C     THERE IS A SLIGHT VARIATION DUE TO THE FACT THAT REDUNDANT COSETS
C     ARE KEPT TRACK OF AND REUSED
C     NCOSE(I)-THE NAME OF A REDUNDANT COSET WHICH CAN BE REUSED
C
      NCO=NCOSET
      DO 9 M=1,NCO
      DO 7 K=1,N3
      INTERM=NREL(K,M)
      IF(INTERM.EQ.0) GO TO 7
      N4ALT=N4(K)-1
      DO 11 L=1,N4ALT
      IIT=R(K,L)
      INTERI=WTAB(INTERM,IIT)
      IF(INTERI.EQ.0) GO TO 13
      INTERM=INTERI
      GO TO 11
   13 IITT=IIT+N5
      IITT=ODD(IITT,NB1)
C
C     IF THERE ARE ANY REUSABLE COSETS THEY ARE USED FIRST
C
      IF(IBET.NE. 0) GO TO 14
      WTAB(INTERM,IIT)=NCOSET+1
      CTIB(INTERM,IIT)=0
C
C     THE NUMBER OF COSETS DEFINED IS INCREASED BY ONE
C
      NCOSET=NCOSET+1
C
C     ROWS ARE CREATED IN THE ARRAY WTAB FOR THE NEW COSET
C
```

```
      WTAB(NCOSET,IITT)=INTERM
      CTIB(NCOSET,IITT)=0
      DO 95 KK=1,N3
      NREL(KK,NCOSET)=NCOSET
   95 CONTINUE
      RETURN
   14 WTAB(INTERM,IIT)=NCOSE(IBET)
      CTIB(INTERM,IIT)=0
      IT=NCOSE(IBET)
      WTAB(IT,IITT)=INTERM
      CTIB(IT,IITT)=0
      DO 17 KK=1,N3
      NREL(KK,NCOSE(IBET))=NCOSE(IBET)
   17 CONTINUE
      NCOSE(IBET)=0
C
C     THE NUMBER OF REUSABLE COSETS IS DECREASED BY ONE
C
      IBET=IBET-1
      RETURN
   11 CONTINUE
    7 CONTINUE
    9 CONTINUE
      EQJ=0
      RETURN
      END


      INTEGER FUNCTION NSUM*2(P1,P2)
C
C     THIS SUBROUTINE MULTIPLIES TWO WORDS TOGETHER
C
      IMPLICIT INTEGER*2(A-Z)
      COMMON /AREA1/ CTAB1(20000),KEEP
      IF(P1.NE.0) GO TO 504
      NSUM=P2
      RETURN
  504 IF(P2.NE.0) GO TO 506
      NSUM=P1
      RETURN
  506 IT=CTAB1(P1)-1
      ITT=CTAB1(P2)-1
      DO 600 I=1,IT
      CTAB1(KEEP+I)=CTAB1(P1+I)
  600 CONTINUE
      KEEP1=KEEP+CTAB1(P1)-1
      DO 602 I=1,ITT
      CTAB1(KEEP1+I)=CTAB1(P2+I)
  602 CONTINUE
      CTAB1(KEEP)=CTAB1(P1)+CTAB1(P2)-1
      NSUM=CHECK(KEEP)
      KEEP=KEEP+CTAB1(KEEP)
      RETURN
      END


      INTEGER FUNCTION CHECK*2(P)
C
C     THIS SUBROUTINE ELIMINATES TRIVIAL RELATORS OF THE FORM XXINVERSE
C     FROM A WORD
```

```
C
      IMPLICIT INTEGER*2(A-Z)
      COMMON N1,N6,N3,N7,NCI,N5,N2(3),N4(1),R(1,7),S(3,6),WTAB(1000,4)
     1,CTIB(1000,4),NREL(5,1000)
      COMMON /AREA1/ CTAB1(20000),KEEP
      JJ=CTAB1(P)
      CHECK=P
      N81=2*N1
  807 IF(CTAB1(P).LE.2) RETURN
      KK=CTAB1(P)-2
      PP=P+1
      PPP=P+2
      DO 800 I=1,KK
      IT=CTAB1(P+I)+N1
      IT=DOD(IT,N81)
      ITT=CTAB1(PP+I)
      IF(IT.NE.ITT) GO TO 800
      IF(KK.EQ.1) GO TO 826
      IF(I.EQ.KK) GO TO 827
      KKT=KK-1
      DO 802 J=I,KKT
      CTAB1(P+J)=CTAB1(PPP+J)
  802 CONTINUE
      CTAB1(P)=CTAB1(P)-2
      IF(CTAB1(P).NE.1)  GO TO 807
      GO TO 826
  800 CONTINUE
      IF(CTAB1(P).EQ.0) CHECK=0
      RETURN
  826 CTAB1(P)=0
      CHECK=0
      RETURN
  827 CTAB1(P+I)=0
      CTAB1(P+I+1)=0
      CTAB1(P)=CTAB1(P)-2
      RETURN
      END


      INTEGER FUNCTION INV*2(P)
C
C     THIS SUBROUTINE GIVES THE INVERSE OF THE WORD INPUTTED
C
      IMPLICIT INTEGER*2(A-Z)
      COMMON N1,N6,N3,N7,NCI,N5,N2(3),N4(1),R(1,7),S(3,6),WTAB(1000,4)
     1,CTIB(1000,4),NREL(5,1000)
      COMMON /AREA1/ CTAB1(20000),KEEP
      IF(P.NE.0) GO TO 502
      INV=0
      RETURN
  502 IT=CTAB1(P)
      N81=2*N1
      CTAB1(KEEP)=IT
      PP=P+IT+1
      KEEP1=KEEP-1
      DO 500 I=2,IT
      JJ=CTAB1(PP-I)+N1
      CTAB1(KEEP1+I)=DOD(JJ,N81)
  500 CONTINUE
      INV=KEEP
```

```
      KEEP=KEEP+CTAB1(P)
      RETURN
      END


      INTEGER FUNCTION DOD*2(X,Y)
C
C     THIS SUBROUTINE GIVES THE VALUE OF X MOD Y
C
      IMPLICIT INTEGER*2(A-Z)
  951 IF(X.GT.Y) GO TO 950
      DOD=X
      GO TO 955
  950 X=X-Y
      GO TO 951
  955 RETURN
      END


$DATA
```

```
 THIS IS THE DATA INPUTTED

THE NUMBER OF ELEMENTS IN THE SUB GROUP GENERATORS
                    1 6 6
THE NUMBER OF ELEMENTS IN THE RELATORS
                    6
THE SUBGROUP GENERATORS
                    1
                    2 1 1 1 1 4
                    4 1 1 1 1 2
THE RELATORS
                    1 1 2 1 4 4


A RELATOR DERIVED FROM REDUNDANCIES. THE FIRST NUMBER IS ONE MORE THAN THE LENGTH OF THE RELATOR.
THE REMAINING NUMBERS ARE THE RELATOR ITSELF
  9      2 3 1 3 4 6 5 5


A RELATOR DERIVED FROM REDUNDANCIES. THE FIRST NUMBER IS ONE MORE THAN THE LENGTH OF THE RELATOR.
THE REMAINING NUMBERS ARE THE RELATOR ITSELF
  7      2 3 1 5 4 6



NUMBER OF COSETS DEFINED          9

REDUNDANT COSETS                      6    9    8    7


THE INDEX OF THE SUBGROUP IS

             5


MTAB-MULTIPLACATION TABLE FOR COSETS
 1:     1     2     1     5
 2:     3     3     5     1
 3:     4     3     2     2
 4:     5     4     3     4
 5:     2     1     4     3
 6:     0     0     0     0
 7:     0     0     0     0
 8:     0     0     0     0
 9:     0     0     0     0



CTAB-POINTERS TO LOCATION OF COSET REPRESENTATIVES
 1:     1     0     3    43
 2:     0     8    13     0
 3:     0    79     0     5
 4:     0   133     0   161
 5:    11    33     0    97
 6:     0     0     0     0
 7:     0     0     0     0
 8:     0     0     0     0
 9:     0     0     0     0
```

```
THE ENTRIES TO TABLE II WHERE THE FIRST NUMBER IS THE LOCATION NUMBER IN THE TABLE ABOVE.
THE SECOND NUMBER IS ONE MORE THAN THE NUMBER OF SYMBOLS IN THE WORD AND THEN THE WORD IS
PRINTED BELOW IN NUMERIC FORM. EACH NUMBER REPRESENTS A SUBGROUP GENERATOR

     1            2
                  1

     3            2
                  4

    43            4
                  4 6 5

     9            3
                  1 1

    13            2
                  5

    79            6
                  2 3 1 6 5

     5            3
                  4 4

   153            8
                  1 1 2 3 1 6 5

   151            8
                  2 3 4 5 5 6 4

    11            2
                  2

    30            4
                  2 3 1

    97            6
                  2 3 4 6 5

CORE USAGE        OBJECT CODE=   23080 BYTES,ARRAY AREA=   67690 BYTES,TOTAL AREA AVAILABLE=   94304  BYTES

DIAGNOSTICS         NUMBER OF ERRORS=       0, NUMBER OF WARNINGS=       0, NUMBER OF EXTENSIONS=       0

COMPILE TIME=     4.70 SEC,EXECUTION TIME=     1.28 SEC,  WATFIV - VERSION 1 LEVEL 3 MARCH  1971      DATE=  72/005
```

7404