Brock Reports in Mathematics and Statistics No. BRMS 200814-01

Jiachao(Jared) Wu

GAP and Its Applications in Combinatorial Group Theory
M.Sc. Project

August 14, 2020

GAP and Its Applications in Combinatorial Group Theory

Jiachao (Jared) Wu

Department of Mathematics and Statistics

Submitted in partial fulfillment
of the requirement for the degree of

Master of Science

Faculty of Mathematics and Science, Brock University
St. Catharines, Ontario

Approved for the Committee:

Supervisor: Dr. Yuanlin Li
Co-Supervisor: Dr. Henryk Fukś

(Department of Mathematics and Statistics)

Dr. Yuanlin Li

Acknowledgments

First and foremost, I would like to express deep and sincere gratitude to my supervisor Dr. Yuanlin Li for giving me the opportunity to do research and providing invaluable guidance and encouragement throughout this research project. During the two-year graduate study, not only I was inspired by his visions, motivations and enthusiasm to mathematics, but also he taught me the methodologies to solve mathematical problems and deal with all kinds of difficulties in the invisible future. Despite of his busy schedule, he was always willing to spending time on me as long as I came by his office with questions or difficulties. I am very thankful to his patience, acceptance and valuable time for weekly discussions and solving all problems during our project. His mathematical thinking and abundant experience are like a compass which can always lead me to a right direction in dealing with a complex mathematical problems. His interesting and easy-understood examples helped me efficiently know well a new abstract concept in mathematics. Thanks to his rigorous logical thinking and strict teaching style, I learned a lot of new knowledge not only in mathematics but also in life throughout the whole project. It is my fortunate and honor to study and do research under his teaching and advising. Additionally, thanks go to his visiting scholar, Chao liu, and I am grateful to Chao Liu, for helping me with programming learning and working with him. As my supervisor and mentor, he taught me more than I could give him credit for here.

Also, I am extending my special thanks to my family for their acceptance, love, sacrifices and support. I am extremely grateful to my supervisors Dr. Henryk Fukś, Dr. Yuanlin Li and the Department of Mathematics and Statistics at Brock University for the financial support and the information technology. I appreciate the opportunity to study and complete my degree at Brock University. Without those support, I cannot image that I would even have a chance to pursue my graduate study and complete such a achievement.

Finally, many thanks go to all the people who have supported me to complete this project directly or indirectly.

Abstract

GAP (Groups, Algorithms, Programming) is system for computational discrete algebra, with particular emphasis on Computational Group Theory. A group G is said to be a $B(n,k)$ group for any n-element subset A of G, $|A^2| \leq k$. In this project, we will apply GAP to some combinatorial problems in group theory, in particular, in characterizations of $B(n,k)$ groups.

In Chapter 1, a brief introduction of GAP is given, which is about its background and its application in mathematics. Also, we give some basic background knowledge and theorems in group theory.

Chapter 2 starts to explore some functionalities of GAP. For each application of a function, an example is provided in details,which shows how to use the function or apply to group theory. Moreover, most of the functions will be used to verifying and characterizing $B(n,k)$ group.

In Chapter 3, we introduce the history and background of $B(n,k)$ group. Next, some GAP programs are applied to the combinatorial group theory. In particular, we complete verifying the characterization of $B(5,19)$ group. In addition, we summarize the differences between characterizations on $B(n,k)$ groups using GAP program and not using GAP.

We continue the investigation of $B(5,k)$-groups, in particular, groups with relations $C_n \rtimes C_4 =< a,b|a^4 = b^n = 1, b^a = b^{-1} >$ in the last chapter. We obtain that All groups $G = C_n \rtimes C_4 =< a,b|a^4 = b^n = 1, b^a = b^{-1} >$ are $B(5,22)$ groups and $B(4,14)$ groups,where $n \geq 11$

and n is an odd number. Additionally, we complete the characterizations on $B(6,23)$ groups. In summary, we provide the characterizations on $B(6,23)$ groups as follows:

Theorem 0.0.1. *If a group is $B(6,23)$ group, then G is one of the following group:*

1) An abelian group;

2) A nonabelian trivial $B(6,23)$ group;

3) $C_3 \rtimes Q_8 =< a,b,c | a^3 = b^4 = c^4, b^2 = c^2, ac = ca, a^b = a^{-1}, b^c = b^3 >$;

4) $C_3 \rtimes (C_4 \times C_2) =< a,b,c | a^3 = b^4 = c^2 = 1, bc = cb, a^b = a^{-1}, ac = ca >$.

Contents

Chapter 1

Introductions

1.1 Introduction of GAP

GAP(Groups, Algorithms, Programming) is an algebra computer system for computational discrete algebra, with particular emphasis on Computational Group Theory. GAP provides mathematical capabilities, which can be accessible through three main ways. The first approach is through a large library of functions, implementing various algebraic algorithms. The second approach is the separate packages of additional functions, which are created by users for specialized purposes. The last one is data library, which contains large classes of algebraic objects. For all algebraic objects are accessible by GAP commands. GAP system contains a programming language, called GAP also. The language can be interpreted, compiled and executed. Users can implement the commands interactively in the GAP prompt or write programs and save as files, which can be compiled and executed. Additionally, GAP language has some excellent features, such as Pascal-like control structures, flexible list and record data types, built-in data type for key algebraic objects, automatic method selection and more. GAP is a free software, which has an open source programming language. Users can use functions directly from GAP data library or their separate packages for specialized purposes. The system(GAP) is distributed freely by GAP users and programmers, including sources. Users can learn and easily extend it for their special needs.

GAP contains a kernel, which is written in C, and it implements the GAP language, memory management and interactive environment for developing and using GAP programs, etc. The GAP system is suitable for any machines with a UNLX-like or recent Windows or MAC OS X operating systems. In our project, we are using the 4.10.2 version of GAP. To install GAP, users can go onto its official website at https:/www.gap-system.org/index.html, where has all the details of installations for different versions or for different operating systems. According to the instructions of the installation, it is easy and straight-forward to complete the installation of GAP and to run GAP by double-clicking on the gapcmd.bat icon. On the website, there is an online manual, which contains plenty of examples of algebraic objects. Also, there is a virtual manual inserted inside the system. The applications of GAP cover many fields in mathematics, such as programming language, the number theory, vectors, modules, groups, rings, finite fields, algebras, combinational structures and more. In our project, we concentrate on applying GAP into group theory and solving combinatorial group problems.

At the GAP prompt, GAP provides an interactive environment, which features line editing(i.e tab completion), break loops for debugging, online help(i.e online access to commands) and a graphical user interface, etc. The interactive environment can always give user instructions and indicate the particular errors for users at the break point. GAP can remember all command history, which contains all commands typed during GAP session, so users can find out all previous executed commands by typing first several letters or sequence. The other way to reuse any previous commands is to press the up-arrow button, thereby, it will return the last command. To find the more previous used command, just repeat the up-arrow. Additionally, data libraries are parts of the core of GAP system, which contains large classes of algebraic objects, such as basic groups, classical groups perfect groups, Integral matrix groups lie algebras and so on. GAP provides a large number of written material, which explains functionality and use of GAP. Last but not least, there is a GAP forum archive, where offers a platform to ask, answer and discuss. Users can always help or get help from each other at the forum .

In chapter 2, we give explanations to some functions of libraries in GAP, which we usually use in group theory. There are examples and explanations for each single topic in group theory or programming language applied. In chapter 3, the application of GAP is to verify B(n,k) groups problems, especially characterization of B(5,19) groups. Some simple instructions of running programs are given as well. Last but not least, we apply GAP program to continue on the studies of the combinatorial group problems in the last chapter. All programs we used in this project are provided in the appendices chapter.

1.2 Preliminary knowledge of group theory

In this section, we list some preliminary knowledge of group theory, which will be used in the later chapters. In the whole project, our notations and terminologies of the group theory follow [17].

Here, we list some groups and their notations for the later chapters. A quanterion group of order 8 is denoted by Q_8, and a dihedral group of order $2n$ is denoted by D_{2n}. Q_8 and D_{2n} can be written as following:

$Q_8 =< a, b|a^4 = 1, a^2 = b^2, a^b = a{-}1 >$

$D_{2n} =< a, b|a^n = b^2 = 1, a^b = a^{-1} >$

The following definitions and theorems will be used in the later chapters.

Definition 1.2.1. *The* ***symmetric group*** S_n *of degree* n *is the group of all permutations on* n *symbols.* S_n *is therefore a permutation group of order* $n!$ *and contains as subgroups every group of order* n*.*

Definition 1.2.2. *A group* G *is called a* ***free group*** *if there exists a generating set* X *of* G *such that every non-empty reduced group word in* X *defines a non-trivial element of* G*.*

Theorem 1.2.3. *(Cayley, 1878). Every group* G *can be imbedded as a subgroup of* S_G *In particular, if* $|G| = n$*, then* G *can be imbedded in* S_n

Theorem 1.2.4. *Let* G *be a nonabelian p-group and* $Z(G)$ *be the center of* G*. Then* G *has two maximal subgroups which contain* $Z(G)$*.*

Definition 1.2.5. *If* H *and* K *are groups, then their direct product, denoted by* $H \times K$*, is the group with elements all ordered pairs (h, k), where* $h \in H$ *and* $k \in K$*, and with operation*

$$(h, k)(h', k') = (hh', kk').$$

Definition 1.2.6. *Let* K *be a (not necessarily normal) subgroup of a group* G*. Then a subgroup* $Q \leqslant G$ *is a* ***complement*** *of* K *in* G *if* $K \cap Q = 1$ *and* $KQ = G$*.*

Definition 1.2.7. *A group* G *is a* ***semidirect product*** *of* K *by* Q*, denoted by* $G = K \rtimes Q$*, if* $K \lhd G$ *and* K *has a complement* $Q_1 \cong Q$*.*

Lemma 1.2.8. *Let Z(G) be the center of G. If G/Z(G) is cyclic, then G is abelian.*

Definition 1.2.9. *Let* G *be a finite group with* n *elements* $a_1, a_2, ..., a_n$*. A* ***multiplication table*** G *is the* $n \times n$ *matrix with* i, j *entry* $a_i * a_j$*:*

G	a_1	a_2	$\cdots$	a_n
a_1	$a_1 * a_1$	$a_1 * a_2$	$\cdots$	$a_1 * a_3$
a_2	$a_2 * a_1$	$a_2 * a_2$	$\cdots$	$a_2 * a_n$
$\cdots$	$\cdots$	$\cdots$	$\cdots$	$\cdots$
a_n	$a_n * a_1$	$a_n * a_2$	$\cdots$	$a_n * a_n$

An important class of groups, called Hamiltonian, is investigated on $B(5, 19)$ groups. We give a definition of Hamiltonian group and a theorem of Hamiltonian group as following.

Definition 1.2.10. *A* ***Hamiltonian group*** *is a non-abelian group in which all subgroups are normal.*

Theorem 1.2.11. *A group H is Hamiltonian if and only if it is the direct product of a quaternion group of order 8, an elementary abelian 2-group and an abelian group with all its elements of odd order.*

The groups of order 24 play an important role in the characterization of $B(5, 19)$ groups. A special linear group, denoted by SL(n,F) of degree n over a field F, is the set of $n \times n$ matrices with determinant 1. A cyclic group of order n is denoted by C_n. The notation of symmetric group of degree n as S_n. A_n is an alternating group of degree n. We list all nonabelian group of order 24 as follows.

1. $C_3 \rtimes C_8 =< a, b | a^3 = b^8 = 1, a^b = a^{-1} >$

2. $SL(2, 3) =< a, b, c | a^3 = b^3 = c^2 = abc >$

3. $C_3 \rtimes Q_8 =< a, b, c | a^3 = b^4 = 1, b^2 = c^2, ac = ca, a^b = a^{-1}, b^c = b^3 >$

4. $C_4 \times S_3 =< a, b, c | a^3 = b^4 = c^2 = 1, bc = cb, ab = ba, a^c = a^{-1} >$

5. $D_{24} =< a^{12} = b^2 = 1, a^b = a^{-1} >$

6. $C_2 \times (C_3 \rtimes C_4) =< a, b, c | a^3 = b^4 = c^2 = 1, bc = cb, a^b = a^{-1}, ac = ca >$

7. $(C_6 \times C_2) \rtimes C_2 =< a, b, c | a^2 = b^2 = c^3 = 1, c^a = c^{-1}, bc = cb, (ba)^4 = 1 >$

8. $C_3 \times D_8 =< a, b, c | a^3 = b^4 = c^2 = 1, b^c = b^{-1}, ab = ba, ac = ca >$

9. $C_3 \times Q_8 =< a, b, c | a^3 = b^4 = 1, b^2 = c^2, ab = ba, ac = ca, b^c = b^{-1} >$

10. $S_4 =< a, b, c | a^2 = b^3 = c^4 = abc = 1 >$

11. $C_2 \times A_4 =< a, b, c | a^3 = b^2 = c^2 = 1, ab = ba, bc = cb, (ca)^3 = 1 >$

12. $C_2 \times C_2 \times S_3 =< a, b, c, d | a^3 = b^2 = c^2 = d^2 = 1, a^b = a^{-1}, c^a = c^{-1}, d^a = d^{-1}, bc = cb, bd = db, cd = dc >$

In chapter 2, we show how to present those groups in GAP. We will discuss some differences of representations between GAP and the samples given above.

Chapter 2

Functions of GAP

In this chapter, we introduce some common-used functions and algorithms in GAP, which are applied in group theory. Most of the functions will be used to solving $B(n, k)$ groups in the later chapter.

2.1 Basic GAP functions in general use

We start with introducing some basic and common-used commands as a programming language. For each command, a particular example would be given at the end of each explanation of the code.

Help manual

Before we introduce functions in GAP, we will show how to use the manual of GAP. On the official web page [19] of GAP, there is an online manual and introduction of GAP. In addition, the manual is also inserted inside the GAP software, so users can search topics by keywords. we can use the help manual interactively through the GAP prompt. After typing "? set" , GAP session will load the list of index entries of "set", and the user can choose a topic to get relative help. "? number" can be used to search concepts or functions, or use down-arrow to choose a topic. Next, an example of query is shown as bellow.

Example 2.1.1.

```
gap> ?sets
    Help: several entries match this topic - type ?2 to get match [2]
    [1] Tutorial: Sets
    [2] Reference: Sets
    [3] Reference: sets
    [4] Reference: Sets of Subgroups
    [5] Reference: setstabilizer
    ......
gap>?2
3.4 Sets
GAP knows several special kinds of lists. A set in GAP is a list
that contains no holes (such a list is called dense) and whose elem
-ents are strictly sorted w.r.t. <; in particular, a set cannot
contain duplicates.
  ....
```

list and List(list)

A list is a collection of objects separated by commas and enclosed in brackets. "List(list)" returns all objects in the list and shows them on the command window. In GAP, There are a lot of operations on list. List is a powerful tool to record and display objects. An example is shown as below:

Example 2.1.2.

```
gap> list:=[1,2,2,3,4,77];
[ 1, 2, 2, 3, 4, 77 ]
gap> list:=["hello","world","!!!"];
[ "hello", "world", "!!!" ]
gap> List(list);
[ "hello", "world", "!!!" ]
```

Some operations on list

List is a powerful tool,which can be used to record or collect objects. Here, we would show some common used functions about list. "Concatenation(list)" or "Concatenation(list1,list2...)" is to combine all objects or lists into one list. There are other two way to connect two or more lists or add more object into the original list. "Add(list, object)" is to add a new object to the list at the tail. "Append(list, list2)" is to add lists at the tail of the original list. The command "Remove(list,position)" is used to remove an element from the list,if there exists such elements. This argument returns the list without the element at the position of the original list. The example is shown as below for those commands mentioned above.

Example 2.1.3.

```
gap> list1:=["hello"];
[ "hello" ]
gap> list2:=["world"];
[ "world" ]
```

```
gap> Concatenation(list1,list2);
[ "hello", "world" ]
gap> Concatenation(last);
"helloworld"
gap> Append(list1,list2);
gap> list1;
[ "hello", "world" ]
gap> Add(list1,list2);
gap> list1;
[ "hello", "world", [ "world" ] ]
gap> Remove(list1,3);
[ "world" ]
gap> list1;
[ "hello", "world" ]
```

Unique(list) and SortedList(list)

List is uesd to store elements and to sort a set of elements in specific order. List can be sorted by calling the command "SortedList(list)". It would arrange order of a list from smallest integer to the largest integer, which are the elements in the list. If the objects in the list are string or word, they would be sorted alphabetically. "Unique(list)" is to eliminate redundant elements in the set. It only keeps the unique element in the list. An example is shown as below.

Example 2.1.4.

```
gap> list:=[1,2,3,2,4,2,3,5];;
```

```
gap> Size(list);

8

gap> Unique(list);

[ 1, 2, 3, 4, 5 ]

gap> SortedList(list);

[ 1, 2, 2, 2, 3, 3, 4, 5 ]

gap> list2:=["q","s","a","b"];

[ "q", "s", "a", "b" ]

gap> SortedList(list2);

[ "a", "b", "q", "s" ]
```

About functions

From above examples, we use the functions to operate on lists. The above functions are from the GAP library. Now, we know how to use functions from the GAP library directly. A GAP function is an expression, which can be written as "function(parameter1, prameter2,..)...end;". In the function, "local" is to create a local variables, which has to be declared before using them. Next, we will show how to create our own functions for solving mathematical problems, and we will have our programs, which are used to solve $B(n, k)$ problems. Here, we show a simple example to solve the problem of Fibonacci sequence by writing functions in GAP language as below.

Example 2.1.5.

```
gap> fib := function ( n )

    >     local f1, f2, f3, i;
```

```
>       f1 := 1; f2 := 1;
>       for i in [3..n] do
>         f3 := f1 + f2;
>         f1 := f2;
>         f2 := f3;
>       od;
>       return f2;
>     end;;
gap> List( [1..10], fib );
[ 1, 1, 2, 3, 5, 8, 13, 21, 34, 55 ]
```

2.2 GAP functions in group theory

permutation

GAP library has a data type, called permutations, to describe the elements of permutation groups. A permutation can be stored in a list of the image of integers. The product of permutation in GAP is slightly different from the some textbook, which is from left to right. For our preference, we still use the direction, which is from right to left, to compute the multiplications of permutations in our project. See example below.

Example 2.2.1.

```
gap> a:=(1,2,3,4);
(1,2,3,4)
gap> (1,2,3,4)*(1,2);
```

```
(2,3,4)
```

Order(G) and Size(list)

Next, the command "Order(G)" returns the order of a group G or the order of an element in the group G. "Size(list)" can be used to count the size of a list or objects of a collection, but the this command cannot compute the order of an element in a group. If using Size(G), it will count the number of elements of the group G. For the purpose to check the order of an element in a group, "Order(a)" can only be applied, otherwise, errors would happen. See example below.

Example 2.2.2.

```
gap> G:=Group([(1,2,3),(1,2)]);; # A group generated by two permutations
gap> Size(G); Order(G); Order((1,2));
6
6
2
```

IsSubgroup(G,S)

In group theory, if we want to check whether a group is a subgroup of a group G, we need to do the subgroup test. In GAP, the command "IsSubgroup(G,S)" is to test if a group S is a subgroup of a group G. GAP session returns the result directly by calling "IsSubgroup(G,s)". See a concrete example below.

Example 2.2.3.

```
gap> g:=Group([(1,2,3,4),(1,2)]);; # S4
```

```
gap> s:=Group([(1,2,3),(1,2)]);;  # S3
gap> IsSubgroup(g,s);   #Test if S3 is a subgroup of S4
true
```

IsNormal(G, N) and NormalSubgroups(G)

Similarly, in group theory, we use normal subgroup test to show if the group N is normal to a group G. In GAP, the command "IsNormal(G,N)" plays the same role as normal subgroup test and returns the result directly. By the way. "NormalSubgroups(G,N)" displays all normal subgroups in group G. See the example as below.

Example 2.2.4.

```
gap> g:=Group([(1,2,3,4),(1,2)]);;
gap> s:=Group([(1,2,3),(1,2)]);
gap> IsNormal(g,s);
false
gap> NormalSubgroups(g);
[ Group([ (1,2,3,4), (1,2) ]), Group([ (2,4,3), (1,4)(2,3), (1,3)(2,4) ]),
Group([ (1,4)(2,3), (1,3)(2,4) ]),  Group(()) ]
```

SylowSubgroup(G, prime)

The following example is find out a Sylow p-subgroup. The result will returns a Sylow p-subgroup is generated by its generators.

Example 2.2.5.

```
gap> g:=SymmetricGroup(4);
Sym( [ 1 .. 4 ] )
gap> SylowSubgroup(g,2);
Group([ (3,4), (1,4)(2,3), (1,3)(2,4) ])
gap> SylowSubgroup(g,3);
Group([ (2,4,3) ])
```

IsomorphismPermGroup(G)

We can find a permutation group P, which is isomorphic to a group G, by calling the command "IsomorphismPermGroup(G)". The result returns a isomorphism from group G to a permutation group. A suitable permutation representation would be selected. We can see an example as below.

Example 2.2.6.

```
gap> g:=SymmetricGroup(4);
Sym( [ 1 .. 4 ] )
gap> i:=IsomorphismPermGroup(g);
IdentityMapping( Group([ (1,2,3,4), (1,2) ]) )
ii:=IsomorphismPermGroup(Image(i));
[ F1, F2, F3, F4 ] -> [ (2,3), (2,3,4), (1,2)(3,4), (1,3)(2,4) ]
```

StructureDescription(G)

In GAP, it has a function to describe structures of a groups. The notation might slightly different from the textbook for some groups, i.e semi-direct of two groups. The command "StructureDescription(G)" would return a string, which tells the structure of a group. For instance, the structure of Group([(1,2,3,4),(1,2)]). In this example, GAP describes Group([(1,2,3,4),(1,2)]) as a symmetric group of degree of 4, which is denoted by S4. An example is shown as follows.

Example 2.2.7.

```
gap> G:=Group([(1,2,3,4),(1,2)]);

Group([ (1,2,3,4), (1,2) ])

gap> StructureDescription(G);

"S4"
```

SymmetricGroup(deg) and its representation

A symmetric group ($S4$) is a typical example. In GAP, we can use the following codes to generate a symmetric group of degree 4.

Example 2.2.8.

```
gap> s4:=SymmetricGroup(4);

Sym( [ 1 .. 4 ] )

gap> g:=Group([(1,2,3,4),(1,2)]);

Group([ (1,2,3,4), (1,2) ])
```

Let a symmetric group of order 24 (or of degree 4) be $S4$. In GAP, there are different representations of $S4$. In GAP, a symmetric group can be generated by the command "SymmetricGroup(degree)". Here, we could interpret the other representation, generated by two permutation groups (1,2,3,4) and (1,2). In GAP, it is written as Group (((1, 2, 3, 4), (1, 2))) or Group ([(1, 2, 3, 4), (1, 2)]). The structure of Group ([(1, 2, 3, 4), (1, 2)]) is $S4$. For the convenience to reader, we include a short proof.

Proposition 2.2.9. *Let G be a group, if $G =< (1,2,3,4),(1,2) >$, then G is a symmetric group of degree 4.*

Proof. Let $G =< (1,2,3,4),(1,2) >$. Since $(1,2,3,4),(1,2) \in S4$, $< (1,2,3,4),(1,2) >$ is a subgroup of $S4$. To show $S4$ is generated by $(1,2,3,4)$ and $(1,2)$, we need to show $|G| = |S4| = 24$. By Lagrange's theorem, since $G \leqslant S4$, $|G|$ can be $1,2,3,4,6,8,12,24$. $|(1,2)| = 2$, $|(1,2,3,4)| = 4$,and $(1,2),(1,2,3,4) \in G$, then we can easily see $|G| \neq 1,2$ or 3. Next, we will show G is not a group of order 6, 8, or 12.

case 1: Assume $|G| = 6$. $(1,2,3,4)\ inG$, but $|(1,2,3,4)| = 4 \nmid G$, which is a contradiction. Hence $G \neq 6$.

case 2: Assume $G = 8$. $(1,2,3,4)(1,2) = (1,3,4)$ (Composition of permutations written in array notation is performed from right to left, that is the permutation on the right is performed first.), and $(1,3,4) \in G$, but $|(1,3,4)| = 3 \nmid 8$, which is a contradiction. Hence, $G \neq 8$.

case 3: Assume $G = 12$. $|G| \leqslant S4$, then $G = A4$. $(1,2) \in G$, which is a odd permutation. This is a contradiction. Hence $G \neq 12$.

Finally, $G = 24$ is the only one possibility. Therefore, $G = S4$. □

Therefore,in GAP, $Group([(1,2,3,4),(1,2)]) = S4 =< (1,2,3,4),(1,2) >$.

FreeGroup and finitely presented group

In GAP, we use a finitely presented group to represent a finite group. Every finite group can be represented as a finitely presented group. Finitely presented group are obtained by factoring a free group by a set of relators. A free group is generated by a finite set of abstract generators. An example is to create a a free group and a finitely presented group as following, which is a dihedral group of order 8.

Example 2.2.10.

```
gap> f:=FreeGroup("a","b");
<free group on the generators [ a, b ]>
gap> g:=f/[f.1^4,f.2^2,f.2*f.1*f.2^-1*f.1];
<fp group on the generators [ a, b ]>
gap> StructureDescription(g);
"D8"
```

SmallGroup Library

In GAP, the SmallGroups library contains large classes of groups. GAP can load all groups of the specific order from the library. We can read a group from the library by calling "SmallGroup(ord,id)". The information of small group with a specific order can show how many groups and what kinds of the groups are. The users can explore all properties of groups by investigating the group of specific order through the SmallGroups library. See example below.

Example 2.2.11.

```
gap> SmallGroupsInformation(24);
  There are 15 groups of order 24.
  They are sorted by their Frattini factors.
     1 has Frattini factor [ 6, 1 ].
     2 has Frattini factor [ 6, 2 ].
     3 has Frattini factor [ 12, 3 ].
     4 - 8 have Frattini factor [ 12, 4 ].
     9 - 11 have Frattini factor [ 12, 5 ].
```

```
    12 - 15 have trivial Frattini subgroup.
  For the selection functions the values of the following attributes
  are precomputed and stored:
    IsAbelian, IsNilpotentGroup, IsSupersolvableGroup, IsSolvableGroup,
    LGLength, FrattinifactorSize and FrattinifactorId.
  This size belongs to layer 2 of the SmallGroups library.
  IdSmallGroup is available for this size.
gap> g:=SmallGroup(24,12);
<pc group of size 24 with 4 generators>
gap> StructureDescription(g);
"S4"
gap> G:=Group([(1,2,3,4),(1,2)]);; StructureDescription(G);
"S4"
gap> IdSmallGroup(G);
[ 24, 12 ]
```

IsomorphismFpGroup(G) and RelatorsOfFpGroup(G)

Since every finite group can be represented by a finitely presented group, GAP performs an isomorphism from a finite group G to a finitely presented group by calling "IsomorphismFpGroup(G)". Generally, this command generate a finitely presented group, which is not the most simplified finitely presented group. GAP can simplify the group into the other isomorphic group, which has less generators. Here, we can use the command "IsomorphicsmSimplifiedFpGroup(G)" to reduce the number of generators. If some generators can be represented by other generators, then the generating set is reducible. "RelatorsOfFpGroup(G)" returns a set of relators of a group G. Each expression of the relators is equal to an identity. For example, ${F_1}^2 = 1$. See the follwing example:

Example 2.2.12.

```
gap> G:=SmallGroup(24,12);
<pc group of size 24 with 4 generators>
gap> iso:=IsomorphismFpGroup(G);
[ f1, f2, f3, f4 ] -> [ F1, F2, F3, F4 ]
gap> RelatorsOfFpGroup(Image(i));
[ F1^2, F2^-1*F1^-1*F2*F1*F2^-1, F3^-1*F1^-1*F3*F1*F4^-1*F3^-1,
 F4^-1*F1^-1*F4*F1*F4^-1*F3^-1, F2^3,  F3^-1*F2^-1*F3*F2*F4^-1*F3^-1,
  F4^-1*F2^-1*F4*F2*F3^-1, F3^2, F4^-1*F3^-1*F4*F3, F4^2 ]
gap> iiso:=IsomorphismSimplifiedFpGroup(Image(iso));
[ F1, F2, F3, F4 ] -> [ F1, F2, F3, F1*F3*F1 ]
gap> RelatorsOfFpGroup(Image(iiso));
[ F1^2, F3^2, F2^3, (F1*F2)^2, F2^-1*F3*F2*F1*F3*F1, (F2^-1*F3)^3 ]
```

From above example, the command "Image(iso)" or "Image(iiso)" is the finitely presented group, which is isomorphic to the group G. As the result shown, the simplified finitely presented group has less generators and much simpler relators than the original group.

MultiplicationTable(G)

Next, the command "MultiplicationTable(G)" returns a square matrix of all positive integers with a range between 1 and the order of the group. In the next chapter, we will use the function to explore the square properties of combinatorial problems. The numbers indicate the positions of elements in the group, which is sorted in a list in GAP. See example below.

Example 2.2.13.

```
gap> g:=Group([(1,2,3),(1,2)]);
Group([ (1,2,3), (1,2) ])
gap> MultiplicationTable(g);
[ [ 1, 2, 3, 4, 5, 6 ],
  [ 2, 1, 4, 3, 6, 5 ],
  [ 3, 5, 1, 6, 2, 4 ],
  [ 4, 6, 2, 5, 1, 3 ],
  [ 5, 3, 6, 1, 4, 2 ],
  [ 6, 4, 5, 2, 3, 1 ] ]
```

To investigate a $B(n,k)$ group, we apply the small groups of squaring property on an n-element subset. In GAP, multiplication table can provide all data of square subset A^2 by selecting any n elements from a group G.

Representations of nonabelian group of order 24

The nonabelian groups of order 24 are listed in chapter 1. Now we will use GAP to present those nonabelian groups. We have a couple of different methods to present all nonablian groups. Here, we will introduce two methods. The first method is listing all nonabelian groups of 24 in the SmallGroup Library, and finding out the isomorphic free groups. The other way is use finitely presented group and factoring out the relators of every generators. In the following example, we will show both methods of representation of some nonablian groups of 24.

Example 2.2.14.

```
This method is to present some of 12 nonabelian groups of order 24 up
 to isomorphism by using the Smallgroup Library in GAP.
The First step is list the ID of nonabelian groups of order 24 in the
SmallGroup library.
gap> Read("Desktop/nonAb.g");
gap> nonabelian(24);
[ [ 24, 1 ], [ 24, 3 ], [ 24, 4 ], [ 24, 5 ], [ 24, 6 ], [ 24, 7 ],
 [ 24, 8 ], [ 24, 10 ], [ 24, 11 ], [ 24, 12 ], [ 24, 13 ], [ 24, 14]]
The next step is to present some nonabelian groups of 24 as below.
The rest will be given in the appendices.
1. [24,3] = C3 : Q8
gap> g3:=SmallGroup(24,4);
<pc group of size 24 with 4 generators>
gap> StructureDescription(g3);
"C3 : Q8"
gap> iso:=IsomorphismFpGroup(g3);
[ f1, f2, f3, f4 ] -> [ F1, F2, F3, F4 ]
gap> isoo:=IsomorphismSimplifiedFpGroup(Image(iso));
[ F1, F2, F3, F4 ] -> [ F1, F2, F1^-2, F4 ]
gap> image:=Image(isoo);
<fp group on the generators [ F1, F2, F4 ]>
gap> RelatorsOfFpGroup(image);
[ F4^3, F2*F1^-1*F2^-1*F1^-1, F2*F1^2*F2, F2*F1^-1*F2*F1,
```

```
F1^-1*F4*F1*F4, F4^-1*F2^-1*F4*F2 ]

2. [24, 6] = D24

gap> g5:=SmallGroup(24,6);

<pc group of size 24 with 4 generators>

gap> StructureDescription(g5);

"D24"

gap> iso:=IsomorphismFpGroup(g5);

[ f1, f2, f3, f4 ] -> [ F1, F2, F3, F4 ]

gap> isoo:=IsomorphismSimplifiedFpGroup(Image(iso));

[ F1, F2, F3, F4 ] -> [ F1, F2, F2^-2, F4 ]

gap> RelatorsOfFpGroup(image);

[ F1^2, F4^3, F2^4, (F1*F4)^2, F2^-1*F1*F2*F1, F4^-1*F2^-1*F4*F2 ]

3. [24, 7] = C2 X (C3 : C4)

gap> g6:=SmallGroup(24,7);

<pc group of size 24 with 4 generators>

gap> StructureDescription(g6);

"C2 X (C3 : C4)"

gap> iso:=IsomorphismFpGroup(g6);

[ f1, f2, f3, f4 ] -> [ F1, F2, F3, F4 ]

gap> isoo:=IsomorphismSimplifiedFpGroup(Image(iso));

[ F1, F2, F3, F4 ] -> [ F1, F2, F1^-2, F4 ]

gap> image:=Image(isoo);

<fp group on the generators [ F1, F2, F4 ]>

gap> RelatorsOfFpGroup(image);
```

```
[ F2^2, F4^3, F1^4, F1^-1*F4*F1*F4, F2*F1^-1*F2*F1, F4^-1*F2*F4*F2 ]
4. [24, 11] = C3 X Q8
gap> g9:=SmallGroup(24,11);
<pc group of size 24 with 4 generators>
gap> StructureDescription(g9);
"C3 X Q8"
gap> iso:=IsomorphismFpGroup(g9);
[ f1, f2, f3, f4 ] -> [ F1, F2, F3, F4 ]
gap> isoo:=IsomorphismSimplifiedFpGroup(Image(iso));
[ F1, F2, F3, F4 ] -> [ F1, F2, F3, F1^-2 ]
gap> image:=Image(isoo);
<fp group on the generators [ F1, F2, F3 ]>
gap> RelatorsOfFpGroup(image);
[ F3^3, F2*F1^-1*F2*F1, F1^-1*F2^-2*F1^-1, F1*F2^-1*F1*F2,
F3^-1*F1^-1*F3*F1, F3^-1*F2^-1*F3*F2 ]
5. [24, 12] = S4
gap> g10:=SmallGroup(24,12);
<pc group of size 24 with 4 generators>
gap> StructureDescription(g10);
"S4"
gap> iso:=IsomorphismFpGroup(g10);
[ f1, f2, f3, f4 ] -> [ F1, F2, F3, F4 ]
gap> isoo:=IsomorphismSimplifiedFpGroup(Image(iso));
[ F1, F2, F3, F4 ] -> [ F1, F2, F3, F1*F3*F1 ]
```

```
gap> image:=Image(isoo);

<fp group on the generators [ F1, F2, F3 ]>

gap> RelatorsOfFpGroup(image);

[ F1^2, F3^2, F2^3, (F1*F2)^2, F2^-1*F3*F2*F1*F3*F1, (F2^-1*F3)^3 ]
```

Example 2.2.15.

This method is used finitely presented groups to present all the nonabelian groups of 24. In this method, we create a free group with the specific number of generators. This example will show some of the nonabelian groups of order 24 as follows.

1. g3 = C3 : Q8

```
gap> f:=FreeGroup("a","b","c");

<free group on the generators [ a, b, c ]>

gap> g3:=f/[f.1^3,f.2^4,f.2^2*f.3^-2, f.1^-1*f.3^-1*f.1*f.3,

f.2^-1*f.1*f.2*f.1,f.3^-1*f.2*f.3*f.2];

<fp group on the generators [ a, b, c ]>

gap> StructureDescription(g3);

"C3 : Q8"

gap> RelatorsOfFpGroup(g3);

[ a^3, b^4, b^2*c^-2, a^-1*c^-1*a*c, b^-1*a*b*a, c^-1*b*c*b ]
```

2. g5 = D24

```
gap> f:=FreeGroup("a","b");

<free group on the generators [ a, b ]>

gap> g5:=f/[f.1^12,f.2^2, f.2^-1*f.1*f.2*f.1];

<fp group on the generators [ a, b ]>
```

```
gap> StructureDescription(g5);
"D24"
gap> RelatorsOfFpGroup(g5);
[ a^12, b^2, b^-1*a*b*a ]
3.  g10 = S4
gap> f:=FreeGroup("a","b","c");
<free group on the generators [ a, b, c ]>
gap> g10:=f/[f.1^2,f.2^3,f.3^4, f.1*f.2*f.3];
<fp group on the generators [ a, b, c ]>
gap> StructureDescription(g10);
"S4"
gap> RelatorsOfFpGroup(g10);
[ a^2, b^3, c^4, a*b*c ]
```

Notice that there are some differences between those two representations. For instance, let us consider D_{24}, which is a dihedral group of order 24. In the first method, we use an isomorphic group in SmallGroup library, which is generated by four abstract generators. After we simplify the finitely presented group, its generating set still has 3 generators. We know a dihedral group from some textbook, which is characterized by the second methods. A dihedral group can be generated by two generators. In general, a finitely presented group is generated by more generators than the regular, which a group is expressed in textbook. For the efficiency, it can reduce a large of calculations in the system using generators with small orders. To discuss the relators of generators, it involves the definitions of generating sets of a group as following:

Definition 2.2.16. *In algebra, **a generating set** of a group is a subset such that every element* *of the group can be expressed as a combination (under the group operation) of finitely many elements of the subset and their inverses.*

Definition 2.2.17. ***Minimal Generating Set:*** *A generating set S for G is a minimal generating set if $S\backslash\{x\}$ is no longer a generating set for G for all $x \in S$.*

Minimal generating set is not unique. For instance, $S_4 =< (1,2,3,4),(1,2) >$ and $S_4 =< a,b,c|a^2 = b^3 = c^4 = abc = 1 >$ are two minimal generating sets of S_4. In GAP, groups in the SmallGroup library is generated by a generating set, but it might not be a minimal generating set. Using more generators in the GAP system is more efficient to handle the computations of groups and saving the computing time. For the convenience of group representation, textbook normally use the minimal generating set to generate a group.

In this chapter, we introduced some common used functions in group theory. We know how to present a group and investigate properties of the group in GAP. When we investigate a group, we can get an isomorphic group from SmallGroup Library, or we can represent the group as a finitely presented group. For the above two methods, there might be a little difference. The representations of the groups are isomorphic to each other, even though the groups are generated by the different number of generators. In general, we investigate a group by investigating its isomorphic group. Next, we will apply GAP into combinatorial problems.

Chapter 3

On $B(n, k)$ groups

In this chapter, we will apply GAP into the B(n,k) group problem. We use GAP programming to show the advantages of solving combinatorial group problems. We will discuss the differences of investigating the combinatorial problems between using computer algebra system and not using computer, in particular, GAP. First, we will give a brief history and background of B(n,k).

3.1 History of $B(n, k)$ groups

Let G be an arbitrary group and A be any n-element subset in G, where $A^2 = \{ab|a, b \in A\}$. To investigate the group G, which satisfies the small squaring property on n-sets, we consider it as a combinatorial problem in group theory. We give some definitions of the small squaring property and B_n-groups as follows.

Definition 3.1.1. *A group G is said to have a **small squaring property** on a n-element subset A of G, if $\{|A^2| < n^2\}$, where $A^2 = \{ab|a, b \in A\}$.*

Definition 3.1.2. *A group G is said to be a B_n**-group** on an n-element subset A, if $\{|A^2| \leq$*

$\frac{n(n+1)}{2}\}$, *where* $A^2 = \{ab|a, b \in A\}$.

Lemma 3.1.3. *If a group is a abelian group, the lowest upper bound for the cardinality of* A^2 *is always less than* $\frac{n(n+1)}{2}$.

Proof. It is easy to be proved. □

Corollary 3.1.4. *If a group* G *is an abelian groups, then* G *is a* B_n*-group.*

This combinatorial problem in group theory started from investing the commutative groups, and research in this subject has a long history. It began with studying B_n-group. Afterwards, some general notations of the small squaring were introduced in [1]. It easy to check that B_n-group has the small squaring property on an n-element subset A in G. Recently, Eddy and Parmenter generalized the notations referred to a new notation as $B(n, k)$ groups in [2].

Definition 3.1.5. *A group* G *is said to be a* ***B(n, k) group*** *if* $A^2 \leq k$ *for any* n*-elements subset* A *of* G *where* n, k *are positive integers with* $k \leq n^2 - 1$ *and* $A^2 = \{ab|a, b \in A\}$.

A group G is called a nontrivial $B(n, k)$ group if $|G| \leq k$. By the definition of $B(n, k)$ group, a B_n-group is a $B(n, \frac{n(n+1)}{2})$ group. Clearly, all abelian groups are $B(n, k)$ groups, since a abelian group is a B_n-group. In general, any groups have the small squaring property on an n-element subset are considered as $B(n, n^2 - 1)$ groups.

It is an interesting problem to determine and classify all $B(n, k)$ groups. Firstly, we summarize which $B(n, k)$ groups have already been characterized till now, where $2 \leq n \leq 7$. When $k \geq \frac{n(n+1)}{2}$, a group G with order $> k$ is nontrivial and nonabelian groups. For $n = 2$ and $n = 3$, the $B(n, k)$ groups have been completely characterized in the literature [1–5]. In [3], a nontrivial and nonabelian group G is a $B(2, 3)$ group if and only if G is a Hamiltonian 2-groups. In [1, 4], they proved that there is no nontrivial nonabelian $B(3, 6)$ group and $B(3, 7)$ group. For $n = 4$,

$B(n,k)$ groups, where $4 \leq k \leq 9$ were characterized in [6]. In [5], Parmenter characterized all $B(4,10)$ groups and he proved there is no nontrivial nonabelian $B(4,10)$ groups. For $k = 11, 12$ or 13, $B(4,k)$ groups were characterized by Li and Tan in [9,10]. Recently, for $k = 14$, $B(4,14)$ groups were characterized by Li and Huang in [12]. For $n = 5$, $B(5,15)$ groups, which is a B_n-group, were characterized by Li and Tan in [8]. $B(5,16)$ groups and $B(5,17)$ groups were characterized by Li and Pan in [6,7]. There is no nontrivial nonabelian $B(5,16)$ groups and $B(5,17)$ groups. Recently, $B(5,18)$ groups were characterized by Li and Huang in [13], and it shows there is no nontrivial nonabelian $B(5,18)$ 2-groups. A nontrivial nonabelian non-2-group is $B(5,18)$ group if and only if $G \cong C_5 \rtimes C_4 =< a, b | a^5 = b^4 = 1, a^b = a^{-1} >$. $B(5,19)$ 2-groups were characterized in [11]. It shows a nontrivial and nonabelian $B(5,19)$ 2-group is isomorphic to a Hamiltonian 2-group.$B(5,19)$ non-2-groups were characterized by Tan and Zhong in [14]. In [16], B_6 and B_7 groups have been completed characterized by Tan and Zhong. They prove there is no nontrivial nonabelian B_6 and B_7 group.

In this chapter, we will apply GAP to verify $B(5,19)$ groups. By GAP programming, we show how to find the subset A, which satisfies the conditions of $B(5,19)$ groups. We will discuss how useful GAP is while solving the combinatorial problem in group theory.

3.2 Applications of GAP On $B(5,19)$ group

Recently, $B(5,19)$ groups has been completely characterized in [11,14]. In those two papers, they applied GAP into solving the B(n,k) groups problems. Now, we continue to explore how powerful and useful GAP is in solving combinatorial group problem. In next section, we compare the methods between $B(5,19)$ and $B(5,18)$ group, which is applying GAP in combinatorial group problems or not.

We list some reasons why we use GAP (a linear computer system) to solve $B(n,k)$ groups problem. Firstly, GAP provides plenty of useful functions and algorithms in group theory. Secondly, GAP has the SmallGroup Library, which can represent every small group. Thirdly, GAP can handle most of calculations in combinatorial group problems efficiently. Compared to Characterization of $B(5,18)$ groups, where calculations are all computed by hand, characterizing $B(5,19)$ groups saves more time on handling computations by using GAP programs, and it is easy to represent a group in the SmallGroup Library. Next, we can explore all the properties of $B(5,19)$ groups. Every finite group can be represented by a finitely presented group. Now, we start with $B(5,19)$ non-2-groups.

On $B(5,19)$ non-2-groups

At first, Tan and Zhong used GAP program [19] to find all nontrivial nonabelian $B(5,19)$ non-2-groups of order up to 257 in the SmallGroups Library in GAP. All the $B(5,19)$ non-2-groups are listed in the Theorem 3.2.12. In GAP, we can find out all the isomorphic groups corresponding to $B(5,19)$ non-2-groups in the SmallGroups Library. For example, SmallGroup(20,1), which has a group structure $C_5 \rtimes C_4$. SmallGroup(20,1) is a nontrivial nonabelian $B(5,19)$ non-2-group. Next, we will use GAP program to show how to verify those groups are $B(5,19)$ groups. In general, we can apply the similar methods to verify any $B(n,k)$ groups. Next, we would introduce how the program find out the $B(5,19)$ group in GAP.

How to find out $B(n,k)$ groups of small orders in GAP

In chapter 2, we have already explained most of the functions in GAP, which we would use in the programs to investigating $B(5,19)$ groups. Here, we explain the idea how we check whether a group is a $B(5,19)$ group. By Corollary 3.1.2, all abelian groups are B_n-groups. For reducing the computing time of the program, first step is to avoid checking all abelian groups. Since non-2-groups and 2–group having different group properties, we divides the test into two parts. One is testing nonabelilan non-2-groups, and the other is testing nonabelian 2-groups. We write a program to check if a group with small order is a $B(5,19)$ group. The program is attached in the appendices A.1.

Next, let us introduce the program in details. In the program, we test every nonabelian group with order m in SmallGroup Library, where $19 < m < 257$. First, we generate a $m \times m$ Matrix by the code "MultiplicationTable(G)", where G is a group with order m to be tested in SmallGroup Library. Next, the program check all 5×5 matrix A^2, which are selected from the $m \times m$ multiplication table. Then, count the number of distinct elements and check if the size of A^2 is less than 19. If any $|A^2| > 19$, then the group is not a $B(5,19)$ group. Then the program continue to check the next A^2. If there is no such A that $|A^2| > 19$, then the group is $B(5,19)$ group. After running the programs and testing all small groups with order less than 257, the results show as the theorem 3.2.12. There are only four nonabelian nontrivial $B(5,19)$ non-2-group. There are infinitely many nonabelian nontrivial $B(5,19)$ 2-groups, which have the structure as $G \cong Q_8 \times E_2$, where E_2 is an elementary 2-groups, and $|E_2| \geq 4$. The nonabelian nontrivial $B(5,19)$ 2-group is isomorphic to a Hamiltonian 2-group. In the next section, we show how to verify those groups are exact $B(5,19)$ groups.

Now, we start with a nonabelian nontrivial $B(5,19)$ group with order 20. A proposition is obtained as below:

Proposition 3.2.1. *If a group $G = C_5 \rtimes C_4 =< a, b | a^5 = b^4 = 1, a^b = a^{-1} >$ is a group, then G is a $B(5,18)$-group. Also, G is $B(5,19)$-group.*

Proof. In [13], Huang and Li completed the formal proof of the characterization of $B(5,18)$-

groups. They show $C_5 \rtimes C_4 =< a, b|a^5 = b^4 = 1, a^b = a^{-1} >$ is a $B(5,18)$-group. Since $C_5 \rtimes C_4 =< a, b|a^5 = b^4 = 1, a^b = a^{-1} >$ is a $B(5,18)$-group, Zhong and Tan mention $C_5 \rtimes C_4 =< a, b|a^5 = b^4 = 1, a^b = a^{-1} >$ is a nonabelian nontrivial $B(5,19)$-group. Here, we will show verifying $C_5 \rtimes C_4 =< a, b|a^5 = b^4 = 1, a^b = a^{-1} >$ is a $B(5,18)$-group by GAP. In GAP, $C_5 \rtimes C_4 =< a, b|a^5 = b^4 = 1, a^b = a^{-1} >$ is represented by SmallGroup(20,1).

We denote SmallGroup(20,1) as G(20,1). Let $H \cong C_5 \rtimes C_4 =< a, b|a^5 = b^4 = 1, a^b = a^{-1} >$. H is isomorphic to $G(20,1)$. We use GAP program to check all possible $|A^2|$ in its multiplication table, where A is any subset of 5 elements in $G(20,1)$. After running the program, the result show that there is no such subset A that $|A^2| > 19$. The program shows the maximal of $|A^2|$ is 18, so $G(20,1)$ is exact $B(5,18)$ group, and $G(20,1)$ is $B(5,19)$ group also. See the program on appendices A.1, which is named B5kk(G).

Next, we explore $G(20,1)$ in GAP. $G(20,1)$ can be represented by a finitely presented group H, where is a fp group of size 20 on the generators $[F_1, F_2, F_3]$. $G(20,1)$ is isomorphic to H. To characterize the group H, we need to find out the relations among F_1, F_2, F_3. Let $H =< F_1, F_2, F_3 >$. By the command RelatorsOfFpGroup(H), we can obtain all relations of $[F_1, F_2, F_3]$. Then, we have $H =< {F_1}^2{F_2}^{-1} = 1, {F_2}^{-1}{F_1}^{-1}F_2F_1 = 1, {F_3}^{-1}{F_1}^{-1}F_3F_1{F_3}^{-3} = 1, {F_2}^2 = 1, {F_3}^{-1}{F_2}^{-1}F_3F_2 = 1, {F_3}^5 >= 1$. From the relations of H, we can conclude $|F_1| = 4$, $|F_2| = 2$ and $|F_3| = 5$. For the convenience to read, we denote F_1, F_2 and F_3 by a,b and c respectively. Then, $H =< a^2b^{-1} = 1, b^{-1}a^{-1}ba = 1, c^{-1}a^{-1}cac^{-3} = 1, b^2 = 1, c^{-1}b^{-1}cb = 1, c^5 = 1 >$. we can see that b commutes with a and c, and $a^2 = b$ since $a^2b^{-1} = 1$. The finitely represented group H can be simplified by the command "IsomorphismSimplifiedFpGroup(H)". After the simplification, the new mapping is $[a, b, c] \to [a, a^2, c]$. The element b can be replaced by a^2. The new relations of H would be $H =< a, c|a^4 = 1, b^5 = 1, a^{-1}ba = b^{-1} >$. H has the group structure, which is $C_5 \rtimes C_4$. Therefore, $H = C_5 \rtimes C_4 =< a, c | a^4 = 1, c^5 = 1, a^{-1}ca = c^{-1} >$. The following codes show how to verify if a group is $B(5,19)$-group and characterize a group. In GAP, the notation ":" represents "$\rtimes$", which is a semi-direct product of two subgroups.

```
gap> Read("Desktop/b5kk.g");
gap> G:=SmallGroup(20,1);
<pc group of size 20 with 3 generators>
gap> StructureDescription(G);
"C5 : C4"
gap> B5kk(G);
[ "The value of k is ", 18 ]
gap> iso:=IsomorphismFpGroup(G);
[ f1, f2, f3 ] -> [ F1, F2, F3 ]
gap> H:=Image(iso);
<fp group of size 20 on the generators [ F1, F2, F3 ]>
gap> RelatorsOfFpGroup(H);
[ F1^2*F2^-1, F2^-1*F1^-1*F2*F1, F3^-1*F1^-1*F3*F1*F3^-3,
  F2^2, F3^-1*F2^-1*F3*F2, F3^5 ]
gap> iso1:=IsomorphismSimplifiedFpGroup(H);
[ F1, F2, F3 ] -> [ F1, F1^-2, F3 ]
gap> H:=Image(iso1);
<fp group on the generators [ F1, F3 ]>
gap> RelatorsOfFpGroup(H);
[ F1^4, F1^-1*F3*F1*F3, F3^5 ]
gap> B5kk(H);
```

```
[ "The value of k is ", 18 ]
```

Next, we verify that $H =< a, c|a^4 = 1, c^5 = 1, a^{-1}ca = c^{-1} >$ is a exact $B(5, 18)$-group. First, we use GAP to list all elements in H. $H = \{1, a, a^3, c, a^3ca, a^2, ac, ca, a^3c, a^2ca, c^2, a^3c^2a, a^2c, aca, ac^2, c^2a, a^3c^2, a^2c^2a, a^2c^2, ac^2a\}$. In GAP, the elements are numbered by their positions in the list. Later, we will use the elements by calling their numbers of the positions in the list. Then, we consider subset A. In this case, H is $B(5, 18)$- group. Therefore, $|A^2|$ must be less than or equal to 18. We try to find A such that $A^2 = 18$. By the function "findElts(H,18)" in appendices A.3, there are 80 combinatorial subsets A in H, where $|A^2| = 18$. Here, we choose two subsets A to testify that $|A^2| = 18$. The first possible subset and the last one are chosen as shown as the following code.

Case 1: A = [2,4,7,10,13]

Let A = [2,4,7,10,13], where the numbers indicate the positions in the list of group G. In the group H, $A = \{a, c, ac, a^2ca, a^2c\}$. We calculate A^2 by hand. Then $A^2 = \{a^2, ac, a^2c, a^3ca, a^3c, ca, c^2, cac, ca^2ca, ca^2c, aca, ac^2, acac, aca^2ca, aca^2c, a^2ca^2, a^2cac, a^2ca^2c, a^2ca^3ca, a^2ca^3c, a^2ca, a^2c^2, a^2cac, a^2ca^2ca, a^2ca^2c\}$. In GAP, it shows the simplified set of A^2. Thus, $A^2 = \{a^2, ac, a^2c, a^3ca, a^3c, ca, c^2, a, a^2c^2a, a^2c^2, aca, ac^2, a^2, a^3c^2a, a^3c^2, c, a^3, c^2, a^2, a, a^2ca, a^2c^2, a^3, c^2a, c^2\}$. After the simplifications, it is easy to eliminate the redundant elements. There are exact 18 distinct element in the set A^2. $A^2 = \{a^2, ac, a^2c, a^3ca, a^3c, ca, c^2, a, a^2c^2a, a^2c^2, aca, ac^2, a^3c^2a, a^3c^2, c, a^3, a^2ca, c^2a\}$. The codes show how we use GAP to determine that a group is exact $B(5, 18)$-group.

Case 2: A = [9, 10, 12, 16, 20]

In group H, $A = \{a^3c, a^2ca, a^3c^2a, c^2a, ac^2a\}$. Thus, $A^2 = \{a^3ca^3c, a^3ca^2ca, a^3ca^3c^2a, a^3c^3a, a^3cac^2a, a^2ca^4c, a^2ca^3ca, a^2ca^4c^2a, a^2cac^2a, a^2ca^2c^2a, a^3c^2a^4c, a^3c^2a^3ca, a^3c^2a^4c^2a, a^3c^2ac^2a, a^3c^2a^2c^2a, c^2a^4c,$

$c^2a^3ca, c^2a^4c^2a, c^2ac^2a, c^2a^2c^2a, ac^2a^3c, ac^2a^3ca, ac^2a^4c^2a, ac^2ac^2a, ac^2a^2c^2a\}$. After the simplification, $A^2 = \{a^2, ac^2a, a^2ca, c^2, ca, a^2c^2, a^2, a^3c^2, a^3ca, ac^2, a^2c^2a, a^3c, c, a, a^2c, a^3c^2a, c, ac, a^2, a^3c, c^2a, ac, a^2c, a^3, c\}$. We sort A^2, then $A^2 = \{\underline{a}, \underline{a^3}, \underline{c}, c, c, \underline{a^3ca}, \underline{a^2}, a^2, a^2, \underline{ac}, ac, \underline{ca}, \underline{a^3c}, \underline{a^3c}, \underline{a^2ca}, \underline{c^2}, \underline{a^3c^2a}, \underline{a^2c}, a^2c, \underline{ac^2}, \underline{c^2a}, \underline{a^3c^2}, \underline{a^2c^2a}, \underline{a^2c^2}, \underline{ac^2a}\}$. The underlined elements are distinct elements in group H. Therefore, there are 18 elements. There is no such A that $|A^2| > 18$. Hence, $< a, c|a^4 = 1, c^5 = 1, a^{-1}ca = c^{-1} >$ is a $B(5,18)$-group.

The codes of the above two cases are shown as following.

```
gap> Read("Desktop/findElt.g");

gap>list:=List(H);

[ <identity ...>, a, a^3, c, a^3*c*a, a^2, a*c, c*a, a^3*c, a^2*c*a,

 c^2, a^3*c^2*a, a^2*c, a*c*a, a*c^2, c^2*a,  a^3*c^2, a^2*c^2*a,

  a^2*c^2, a*c^2*a ]

gap> findElts(H,18);

[ 2, 4, 7, 10, 13 ]

....# 78 possible subsets A

[ 9, 10, 12, 16, 20 ]

gap># case 1

gap> A:=[2,4,7,10,13];

[2,4,7,10,13]

gap> list{A};

[ a, c, a*c, a^2*c*a, a^2*c ]

gap> m:=MultiplicationTable(g);;

gap> A2:=m{A}{A};

[ [ 6, 7, 13, 5, 9 ], [ 8, 11, 2, 18, 19 ], [ 14, 15, 6, 12, 17 ],
```

```
[ 4, 3, 11, 6, 2 ], [ 10, 19, 3, 16, 11 ] ]

gap> A2:=Concatenation(A2);

[ 6, 7, 13, 5, 9, 8, 11, 2, 18, 19, 14, 15, 6, 12, 17,

 4, 3, 11, 6, 2, 10, 19, 3, 16, 11 ]

gap> A2:=list{A2};

[ a^2, a*c, a^2*c, a^3*c*a, a^3*c,

 c*a, c^2, a, a^2*c^2*a, a^2*c^2,

 a*c*a, a*c^2, a^2, a^3*c^2*a, a^3*c^2,

 c, a^3,  c^2, a^2, a,

 a^2*c*a, a^2*c^2, a^3, c^2*a, c^2 ]

gap> SortedList(A2);

[ a, a, a^3, a^3, c, a^3*c*a, a^2, a^2, a^2, a*c, c*a, a^3*c, a^2*c*a,

c^2,c^2, c^2,a^3*c^2*a, a^2*c, a*c*a, a*c^2,  c^2*a, a^3*c^2, a^2*c^2*a,

 a^2*c^2, a^2*c^2 ]

gap> Unique(A2);

[ a^2, a*c, a^2*c, a^3*c*a, a^3*c, c*a,

 c^2, a, a^2*c^2*a, a^2*c^2, a*c*a, a*c^2,

 a^3*c^2*a, a^3*c^2, c, a^3, a^2*c*a, c^2*a ]

gap> Size(Unique(A2));

18

gap> #Case 2

gap> A:=[ 9, 10, 12, 16, 20 ];

[ 9, 10, 12, 16, 20 ]

gap> list{A};
```

```
[ a^3*c, a^2*c*a, a^3*c^2*a, c^2*a, a*c^2*a ]
gap> A2:=m{A}{A};
[ [ 6, 20, 10, 11, 8 ], [ 19, 6, 17, 5, 15 ], [ 18, 9, 4, 2, 13 ],
[ 12, 4, 7, 6, 9 ], [ 16, 7, 13, 3, 4 ] ]
gap> A2:=Concatenation(A2);
[ 6, 20, 10, 11, 8, 19, 6, 17, 5, 15, 18, 9, 4, 2, 13 ,
 12, 4, 7, 6, 9, 16, 7, 13, 3, 4 ]
gap> A2:=list{A2};
[ a^2, a*c^2*a, a^2*c*a, c^2, c*a, a^2*c^2, a^2, a^3*c^2, a^3*c*a, a*c^2,
 a^2*c^2*a, a^3*c, c, a, a^2*c, a^3*c^2*a,  c, a*c, a^2, a^3*c, c^2*a,
  a*c, a^2*c, a^3, c ]
gap> SortedList(A2);
[ a, a^3, c, c, c, a^3*c*a, a^2, a^2, a^2, a*c, a*c, c*a, a^3*c, a^3*c,
 a^2*c*a, c^2, a^3*c^2*a, a^2*c, a^2*c, a*c^2,  c^2*a, a^3*c^2,
  a^2*c^2*a, a^2*c^2, a*c^2*a ]
gap> Unique(A2);
[ a^2, a*c^2*a, a^2*c*a, c^2, c*a, a^2*c^2, a^3*c^2, a^3*c*a, a*c^2,
a^2*c^2*a, a^3*c, c, a, a^2*c, a^3*c^2*a, a*c,  c^2*a, a^3 ]
gap> Size(Unique(A2));
18
```

□

Proposition 3.2.2. *If a group G has a group structure $C_5 \rtimes C_4$, then G is not necessary a $B(5,19)$-group.*

Proof. From theorem 3.2.12, $C_5 \rtimes C_4 =< a, b | a^4 = 1, b^5 = 1, a^{-1}ba = b^{-1} >$ is a $B(5,18)$-group.

From SmallGroup library in GAP, we know there are 5 groups of order 20. There are two groups which have the same group structure $C_5 \rtimes C_4$. Here, we prove by contradiction. Consider $C_5 \rtimes C_4 =< a, b|a^4 = 1, b^5 = 1, a^{-1}ba = b^2 >$. Let G be $C_5 \rtimes C_4 =< a, b|a^4 = 1, b^5 = 1, a^{-1}ba = b^2 >$, and $G = \{1, a, a^{-1}, b, b^{-1}, a^2, ab, ab^{-1}, a^{-1}b, a^{-1}b^{-1}, ba, ba^{-1}, b^2, b^{-1}a, b^{-1}a^{-1}, b^{-2}, a^2b, a^2b^{-1}, aba, ab^{-1}a\}$. Suppose G is $B(5,19)$-group. For every suset A in G, $|A^2| \leq 19$. Let $A = \{1, a, b, ba, a^2b\}$, Thus, $A^2 = \{1, a, b, ba, a^2b, a, a^2, ab, aba, a^3b, b, ba, b^2, b^2a, ba^2b, ba, ba^2, bab, baba, ba^3b, a^2b, a^2ba, a^2b^2, a^2b^2a, a^2ba^2b\}$. Since $a^{-1}ba = b^{-2}$, we can simplify A^2. We can get the simplification of A^2 from GAP. Thus, $A^2 = \{\underline{1}, \underline{a}, \underline{b}, \underline{ba}, \underline{a^2b}, a, \underline{a^{-1}}, \underline{a^2}, \underline{b^2}, \underline{ba^{-1}}, b, \underline{b^{-1}a}, \underline{ab^{-1}}, \underline{ab}, \underline{b^{-1}}, ba, \underline{a^{-1}b}, \underline{b^{-1}a^{-1}}, 1, \underline{a^{-1}b^{-1}}, a^2b, \underline{b^{-2}}, \underline{ab^{-1}a}, \underline{a^2b^{-1}}, \underline{aba}\}$. We can easily see that there are 20 distinct underlined elements in A^2, which is a contradiction. Therefore, G is not a $B(5,19)$-group.

Hence, If a group G has a group structure as $C_5 \rtimes C_4$, then G is not necessary a $B(5,19)$-group.

Some GAP applications are attached below.

```
gap> f:=FreeGroup("a","b");
<free group on the generators [ a, b ]>
gap> G:=f/[f.1^4,f.2^5,f.1^-1*f.2*f.1*f.2^-2];
<fp group on the generators [ a, b ]>
gap> StructureDescription(G);
"C5 : C4"
gap> Read("Desktop/B5kk.g");
gap> B5kk(G);
[ "The value of k is ", 20 ]
gap> Read("Desktop/findElt.g");
gap> A:=[ 1, 2, 4, 11, 17 ];;
gap> list:=List(G);
```

```
[<identity ...>, a, a^-1, b, b^-1, a^2, a*b, a*b^-1, a^-1*b, a^-1*b^-1,

b*a, b*a^-1, b^2, b^-1*a, b^-1*a^-1,b^-2,a^2*b,a^2*b^-1,a*b*a, a*b^-1*a]

gap> m:=MultiplicationTable(G);;

gap> A2:=m{A}{A};

[ [ 1, 2, 4, 11, 17 ], [ 2, 3, 6, 13, 12 ], [ 4, 14, 8, 7, 5 ],

[ 11, 9, 15, 1, 10 ], [ 17, 16, 20, 18, 19 ] ]

gap> A2:=Concatenation(A2);

[ 1, 2, 4, 11, 17, 2, 3, 6, 13, 12, 4, 14, 8, 7, 5,

 11, 9, 15, 1, 10, 17, 16, 20, 18, 19 ]

gap> list{A2};

[ <identity ...>, a, b, b*a, a^2*b, a, a^-1, a^2, b^2, b*a^-1, b,

 b^-1*a, a*b^-1, a*b, b^-1, b*a, a^-1*b, b^-1*a^-1,  <identity ...>,

  a^-1*b^-1, a^2*b, b^-2, a*b^-1*a, a^2*b^-1, a*b*a ]

gap> SortedList(A2);

[ 1, 1, 2, 2, 3, 4, 4, 5, 6, 7, 8, 9, 10, 11, 11, 12, 13, 14, 15,

16, 17, 17, 18, 19, 20 ]

gap> Size(Unique(A2));

20
```

□

We complete verifying $B(5,19)$ groups of order 20. Next, we analyze the other nonabelian $B(5,19)$ groups of order 24. There are three nonabelian and nontrivial $B(5,19)$-groups of order of 24. The following proposition is obtained:

Proposition 3.2.3. *If a group G is a nonabelian nontrivial $B(5,19)$-group of order 24, then G is one of the following groups.*

(1) $C_3 \rtimes Q_8 =< a,b,c | a^3 = b^4 = 1, b^2 = c^2, ac = ca, a^b = a^{-1}, b^c = b^3 >$

(2) $C_3 \rtimes (C_4 \times C_2) =< a,b,c | a^3 = b^4 = c^2 = 1, bc = cb, a^b = a^{-1}, ac = ca >$

(3) $C_3 \times Q_8 =< a^4 = b^4 = c^3 = 1, b^a = b^3, a^b = a^3, ac = ca, c^b = c >$

Proof. In chapter 2, we can list all nonabelian groups from SmallGroup Library in GAP. There are 12 nonabelian groups of order 24, which we found in GAP. And the ID of those groups in GAP are [24, 1], [24, 3], [24, 4], [24, 5], [24, 6], [24, 7], [24, 8], [24, 10], [24, 11], [24, 12], [24, 13], [24, 14]. Next, Let G be a nonabelian group of 24 in SmallGroup library, for example, G = SmallGroup(24,1). We denote SmallGroup(ord,id) as G(ord, i), where ord is the order of group G and i is the ID of the group in the SmallGroup Library. By calling the function "B5k(G,19)", the output will display the result that is "["The value of k is ", n]". If the value of k is less than or equal to 19, then the group G is a $B(5,19)$ group. Otherwise, G is not a $B(5,19)$ group. After testing all nonabelian groups of order 24, there are 3 nonabelian $B(5,19)$ groups of order 24, which are $G(24,4)$, $G(24,7)$ and $G(24,11)$. As we present the nonabelian groups of 24 in chapter 2 or in the appendices, we conclude that the representations of the three nonableian $B(5,19)$ groups as follows:

(1) $G(24,4) = C_3 \rtimes Q_8 =< a,b,c | a^3 = b^4 = 1, b^2 = c^2, ac = ca, a^b = a^{-1}, b^c = b^3 >$

(2) $G(24,7) = C_3 \rtimes (C_4 \times C_2) =< a, b, c | a^3 = b^4 = c^2 = 1, bc = cb, a^b = a^{-1}, ac = ca >$

(3) $G(24,11) = C_3 \times Q_8 =< a^4 = b^4 = c^3 = 1, b^a = b^3, a^b = a^3, ac = ca, c^b = c >$

Next, we discuss those three groups separately by using GAP program. we remark that those three groups are not $B(5,18)$ groups.

Case 1: G(24,4)

We use GAP to find out all subsets A, which $|A^2| = 19$. We will choose one of the subsets to discuss. In this case, there are 1272 possible subsets A, which $|A^2| = 19$. Here, for the convenience to read, we use the representation to present $G(24,4)$, which is $C_3 \rtimes Q_8 =< a, b, c | a^3 = b^4 = 1, b^2 = c^2, ac = ca, a^b = a^{-1}, b^c = b^3 >$

```
gap> f:=FreeGroup("a","b","c");;
gap> G:=f/[f.1^3,f.2^4,f.3^4,f.2^2*f.3^-2,f.1*f.3*f.1^-1*f.3^-1,
f.3^-1*f.2*f.3*f.2,f.2^-1*f.1*f.2*f.1];;
gap> StructureDescription(G);
"C3 : Q8"
gap> findElts(G,19);
[ 2, 3, 5, 6, 11 ]
[ 2, 3, 5, 6, 19 ]
......
[ 16, 17, 20, 21, 23 ]
[ 17, 20, 21, 22, 23 ]
There are 1272 subset A, which |A^2| = 19.
gap> A:=[2,3,5,6,11];
gap> list:=List(G);
```

```
[ <identity ...>, a, a^-1, b, b^-1, c, c^-1, a*b, a*b^-1, a*c,
 a*c^-1, a^-1*b, a^-1*b^-1, a^-1*c, a^-1*c^-1, b^2,  b*c, b*c^-1,
  a*b^2, a*b*c, a*b*c^-1, a^-1*b^2, a^-1*b*c, a^-1*b*c^-1 ]
gap> list{A};
[ a, a^-1, b^-1, c, a*c^-1 ]
gap> m:=MultiplicationTable(G);;
gap> A2:=m{A}{A};
[ [ 4, 6, 8, 9, 15 ], [ 13, 4, 10, 2, 17 ], [ 16, 10, 12, 21, 19 ],
 [ 3, 7, 14, 4, 20 ], [ 22, 17, 19, 24, 12 ] ]
gap> C:=Concatenation(A2);;
gap> list{C};
[ b, c, a*b, a*b^-1, a^-1*c^-1, a^-1*b^-1, b, a*c, a, b*c, b^2, a*c,
 a^-1*b, a*b*c^-1, a*b^2, a^-1, c^-1, a^-1*c, b,  a*b*c, a^-1*b^2,
  b*c, a*b^2, a^-1*b*c^-1, a^-1*b ]
gap> Size(Unique(C));
19
```

Case 2: G(24,7)

In this case, there are 192 possible Subsets A, which $|A^2| = 19$. We will choose the representation to present $G(24,7)$, which $G(24,7) = C_3 \rtimes (C_4 \times C_2) =< a,b,c|a^3 = b^4 = c^2 = 1, bc = cb, a^b = a^{-1}, ac = ca >$.

```
gap> f:=FreeGroup("a","b","c");;
gap> G:=f/[f.1^3,f.2^4,f.3^2,f.2*f.3*f.2^-1*f.3^-1,
f.1*f.3*f.1^-1*f.3^-1,f.2^-1*f.1*f.2*f.1];;
gap> StructureDescription(G);
```

```
"C2 x (C3 : C4)"
gap> findElts(G,19);
[ 3, 5, 9, 14, 16 ]
[ 3, 5, 9, 14, 22 ]
......
[ 15, 17, 19, 21, 24 ]
[ 15, 18, 19, 20, 23 ]
There are 192 subsets, which |A^2| = 19.
gap> A:=[3,5,9,14,16];;
gap> list:=List(G);
[ <identity ...>, a, a^-1, b, b^-1, c, a*b, a*b^-1, a*c, a^-1*b,
 a^-1*b^-1, a^-1*c, b^2, b*c, b^-1*c, a*b^2, a*b*c,  a*b^-1*c,
  a^-1*b^2, a^-1*b*c, a^-1*b^-1*c, b^2*c, a*b^2*c, a^-1*b^2*c ]
gap> list{A};
[ a^-1, b^-1, a*c, b*c, a*b^2 ]
gap> m:=MultiplicationTable(G);;
gap> A2:=m{A}{A};
[ [ 4, 10, 1, 15, 21 ], [ 18, 12, 23, 6, 2 ], [ 1, 17, 4, 8, 24 ],
 [ 22, 21, 16, 4, 3 ], [ 14, 2, 20, 18, 5 ] ]
gap> C:=Concatenation(A2);;
gap> list{C};
[ b, a^-1*b, <identity ...>, b^-1*c, a^-1*b^-1*c, a*b^-1*c, a^-1*c,
 a*b^2*c, c, a, <identity ...>, a*b*c, b, a*b^-1,  a^-1*b^2*c, b^2*c,
  a^-1*b^-1*c, a*b^2, b, a^-1, b*c, a, a^-1*b*c, a*b^-1*c, b^-1 ]
```

```
gap> Size(Unique(C));
19
```

Case 3: G(24,11)

In this case, there are 1080 possible Subsets, which $|A^2| = 19$. We will choose the representation to present $G(24,11)$, which $G(24,11) = C_3 \times Q_8 =< a^4 = b^4 = c^3 = 1, b^a = b^3, a^b = a^3, ac = ca, c^b = c >$

```
gap> f:=FreeGroup("a","b","c");;
gap> G:=f/[f.1^3,f.2^4,f.2^2*f.3^-2,f.1*f.3*f.1^-1*f.3^-1,
f.1*f.2*f.1^-1*f.2^-1,f.3^-1*f.2*f.3*f.2];;
gap> StructureDescription(G);
"C3 x Q8"
gap> findElts(G,19);
[ 2, 3, 6, 7, 22 ]
[ 2, 3, 6, 9, 23 ]
......
[ 14, 18, 20, 21, 22 ]
[ 14, 20, 21, 22, 23 ]
There are 1080 subsets, which |A^2| = 19.
gap> A:=[2,3,6,7,22];;
gap> list:=List(G);
[ <identity ...>, a, a^-1, b, b^-1, c, c^-1, a*b, a*b^-1, a*c, a*c^-1,
 a^-1*b, a^-1*b^-1, a^-1*c, a^-1*c^-1, b^2, b*c, b*c^-1, a*b^2, a*b*c,
  a*b*c^-1, a^-1*b^2, a^-1*b*c, a^-1*b*c^-1 ]
gap> list{A};
```

```
[ a, a^-1, c, c^-1, a^-1*b^2 ]
gap> m:=MultiplicationTable(G);;
gap> A2:=m{A}{A};
[ [ 5, 6, 10, 12, 11 ], [ 14, 5, 2, 21, 20 ], [ 3, 8, 5, 9, 23 ],
  [ 12, 13, 18, 19, 1 ], [ 11, 24, 17, 1, 12 ] ]
gap> C:=Concatenation(A2);;
gap> list{C};
[ b^-1, c, a*c, a^-1*b, a*c^-1, a^-1*c, b^-1, a, a*b*c^-1, a*b*c,
  a^-1,a*b, b^-1, a*b^-1, a^-1*b*c, a^-1*b, a^-1*b^-1, b*c^-1, a*b^2,
   <identity ...>, a*c^-1, a^-1*b*c^-1, b*c, <identity ...>, a^-1*b ]
gap> Size(Unique(C));
19
```

From above three cases, we use GAP check all subsets A in each group. There are no such subsets A that $|A^2| > 19$, and there are certain amount of subsets A, which $|A^2| = 19$. We can call those three groups exact $B(5,19)$ groups, and they are not $B(5,18)$ groups. To verify the size of A^2, select a subset A as shown in the example. For example, in case 1, we choose $A = [2,3,5,6,11]$, which represents that 5 elements from the respective positions in the given group, and the five elements are $a, a^{-1}, b^{-1}, c, ac^{-1}$. In GAP, all elements in square subset A^2 can be checked if they are a unique element in the set. After checking the uniqueness, there are exactly 19 distinct elements in the set A^2 in this case. □

With using GAP program, we check all nonabelian non-2-groups up to 257 in the Small-Groups Library, there are four nonabelian $B(5,19)$ non-2-groups as mentioned above. Tan and Zhong proved the follwing propositions in [14].

Proposition 3.2.4. *Let G be a $B(5,19)$ non-2-group of order > 257. Then G is a $B(4,14)$ group.*

Proposition 3.2.5. *Let G be a $B(5,19)$ non-2-groups of order ≥ 257. Then G must be abelian.*

The details of the proof can be viewed in [14].

On $B(5,19)$ 2-groups

In [11], Moss, Wang and Tan completed the characterization of $B(5,19)$ 2-groups with using GAP program. The groups of order 32 play an important role in the characterization of $B(5,19)$ 2-groups. Fisrt, they check all nonabelian 2-groups of order 32, the following lemma is obtained:

Lemma 3.2.6. *$Q_8 \times C_2 \times C_2$ is the only nonabelian $B(5,19)$ group of order 32. Moreover, it is not a $B(5,18)$ group.*

The following lemma is required in the proof of propositions later.

Lemma 3.2.7. *Theorem 3.9 [9]. $Q_8 \times E_2$ are $B(4,12)$ groups.*

Next, they prove following propositions:

Proposition 3.2.8. *$Q_8 \times E_2$ are $B(5,19)$ groups.*

Proposition 3.2.9. *Let G be a nontrivial nonabelian $B(5,19)$ 2-group. Then G has a nonabelian maximal subgroup.*

Proposition 3.2.10. *If G is a nontrivial nonabelian $B(5,19)$ 2-group, then G is isomorphic to a Hamiltonian 2-group.*

A main result is obtained as the following theorem:

Theorem 3.2.11. *G is a nontrivial nonabelian $B(5,19)$ 2-group if and only if $G \cong Q_8 \times E_2$, where E_2 is an elementary 2-groups with $|E_2| \geq 4$.*

For more details of proofs, see [11].

Combined the main results of [11,14], $B(5,19)$ groups has been completed characterizations with using GAP program. The result is shown as the following theorem.

Theorem 3.2.12. *Let G be a nontrivial nonabelian $B(5,19)$ group. Then G is one of the following groups.*

(1) $C_5 \rtimes C_4 =< a, b|a^4 = 1, b^5 = 1, a^{-1}ba = b^{-1} >$, $C_3 \rtimes Q_8$, $C_3 \rtimes (C_4 \times C_2)$, $C_3 \times Q8$

(2) $G \cong Q_8 \times E_2$, *where* $|E_2| \geq 4$.

3.3 comparisons on B(5,18) and B(5,19) groups

In this section, we compare the approaches of characterizations on $B(n,k)$ groups. In particular, we deal with $B(5,18)$ groups, which does not use GAP, and $B(5,19)$ groups, which applys GAP program. Next, let us start with $B(5,18)$ groups.

On $B(5,18)$ groups

From the previous section, we know how to characterize $B(5,19)$ groups with applying GAP. It divides into two parts. One is on $B(5,19)$ non-2-groups, and the other is $B(5,19)$ 2-groups. Characterizations of $B(5,18)$ groups is also divided into two parts, which are $B(5,18)$ non-2-groups and 2-groups. Next, we briefly introduce how to characterize $B(5,18)$ groups without using GAP program in [13].

On $B(5,18)$ non-2-groups

To characterize $B(5,18)$ non-2-groups, we follow the steps as below:

Step 1: In order to find the necessary condition, we need to prove the 6 following lemmas.

Lemma 3.3.1. *Let P be a Sylow subgroup of odd order of a $B(5,20)$ group G. Then P is abelian. In particular, if G is a $B(5,18)$ group, then P is abelian.*

Lemma 3.3.2. *Let G be a $B(5,18)$ group of odd order. Then G is abelian.*

Lemma 3.3.3. *Let G be a nontrivial $B(5,18)$ non-2-group with a nontrivial Sylow 2-subgroup P. Then G has a normal subgroup T of odd order such that G=TP.*

Lemma 3.3.4. *T is abelian and not centralized by P.*

Lemma 3.3.5. *P has a subgroup Q of index 2 which centralizes T and every element of $P - Q$ inverts T.*

Lemma 3.3.6. *P is abelian, and the exponent of Q is at most 2.*

Step 2: After we prove the above six lemma, we get the necessary condition as following theorem:

Theorem 3.3.7. *Let G be a nontrivial nonabelian $B(5,18)$ non-2-group. Then $G = TP$, where T is a normal abelian subgroup of a odd order and P is a nontrivial abelian Sylow 2-subgroup of G. Furthermore, the subgroup $Q = C_P(T)$ has index 2 in P, the exponent of Q is at most 2, and each element of $P - Q$ inverts T.*

In order to complete characterization, we need the 2 following lemmas.

Lemma 3.3.8. *Let $G \cong < a, b | a^5 = b^4 = 1, a^b = a^{-1} >$. Then G is a $B(5,18)$ group.*

Lemma 3.3.9. *D_{2n} with $n \geq 10$ is not a $B(5,18)$ group.*

Now, a complete characterization of $B(5,18)$ non-2-groups is given as below.

Theorem 3.3.10. *A non-2-group G is a $B(5,18)$ group if and only if one of the following statements holds:*

(1) G is abelian;

(2) G is a trivial $B(5,18)$ group;

(3) $G \cong < a, b | a^5 = b^4 = 1, a^b = a^{-1} >$.

Corollary 3.3.11. *If a group G is a nonabelian and nontrivial $B(5,18)$ group, then $G \cong < a, b | a^5 = b^4 = 1, a^b = a^{-1} >$.*

From the above steps, we need to prove all lemmas and theorems to complete the characterization on $B(5,18)$ non-2-groups. All the proofs of lemmas and propositions are independent. For each proof, there are many cases to be considered, which contain plenty of tedious computations calculated by hand. Hence, the method of characterization on $B(5,18)$ non-2-groups are complicated, and there are a lot of cases to be analyzed for each proof. At the beginning of this project, we tried using the same method on $B(5,,18)$ non-2-groups to solve $B(5,19)$ non-2-groups, but it is much more complicate than on $B(5,18)$ non-2-groups. During some cases analyzing, we cannot find out a suitable subset A to complete a proof. We are not sure if we can characterize $B(5,19)$ non-2-groups with using the same approach on $B(5,18)$ non-2-groups. Compared with characterization on $B(5,18)$ non-2-groups, we adopt a new approach on characterization on $B(5,19)$ non-2-group with using GAP. For $B(5,19)$ non-2-groups, we prove that a group is $B(5,19)$ non-2-groups of order ≥ 257 is a $B(4,14)$ group or a abelian group. For $B(5,19)$ non-2-groups of order < 257, we apply GAP programs to find out all non-abelian non-2-groups, which are $C_5 \rtimes C_4 =< a, b | a^5 = b^4 = 1, a^b = a^{-1} >$,$< a, b | a^5 = b^4 = 1, a^b = a^{-1} >$, $C_3 \rtimes Q_8$, $C_3 \rtimes (C_4 \times C_2)$ and $C_3 \times Q8$. This approach can also be applied to characterize $B(5,18)$ non-2-groups, and it is much more efficient.

The following table is showing main differences of characterizations on $B(5,18)$ and $B(5,19)$

non-2-groups.

	$B(5,18)$ non-2-groups	$B(5,19)$ non-2-groups
Using GAP	no	yes
nonabelian nontrivial $B(n,k)$ non-2-groups	$C_5 \rtimes C_4 =< a, b \vert a^5 = b^4 = 1, a^b = a^{-1} >$	$< a, b \vert a^5 = b^4 = 1, a^b = a^{-1} >$, $C_3 \rtimes Q_8$, $C_3 \rtimes (C_4 \times C_2)$, $C_3 \times Q8$
Computation	by hand	by GAP
proofs	more	less

On $B(5,18)$ 2-groups

To investigate $B(5,18)$ 2-groups, first we assume every proper subgroup of G is a abelian. Then G is a trivial $B(5,18)$ 2-group. Next, consider groups of order 32. All maximal subgroups of nonabelian $B(5,18)$ 2-groups are abelian. The 2-groups of order 16 play an important role in characterizing $B(5,18)$ 2-groups. There are 9 nonabelian groups of order 16. After testing all nonabelian groups of order 32, we found there is no nontrivial nonabelian $B(5,18)$ 2-group. By using minimal counterexample method, we can prove there is no nontrivial nonabelian $B(5,18)$ 2-groups.

For $B(5,19)$ 2-groups, we used a similar approach, and we consider all nonabelian groups of order 32. The difference are $B(5,18)$ 2-group analyze all cases by hand calculation, but we use GAP program to check all nonabelian group of order 32 if they are $B(5,19)$ 2-groups. Additionally, $B(5,19)$ 2-group used induction to prove that nonabelian nontrivial $B(5,19)$ 2-groups are isomorphic Hamiltonian-2-groups.

The main comparison is given is the following table:

	$B(5,18)$ 2-groups	$B(5,19)$ 2-groups
Method	minimal counterexample method	induction
Considering groups	groups of 32	
Important role	groups of 16	groups of order 32
Using GAP	no	yes
Result	no nonabelin $B(5,18)$ groups order 32	$Q_8 \times C_2 \times C_2$
Proof	cases analyzing	GAP programs
Nonablian $B(n,k)$ groups	no	Hamiltonian 2-groups
Computation	by hand	by GAP

Combining the results of $B(5,18)$ non-2-groups and $B(5,18)$ 2-groups, the following theorem is obtained:

Theorem 3.3.12. *A group G is a $B(5,18)$ group if and only if one of the following statements holds:*

(1) G is abelian;

(2) G is a trivial $B(5,18)$ group;

(3) $G \cong < a, b | a^5 = b^4 = 1, a^b = a^{-1} >$.

To view all proofs of the above lemmas and theorems, all details are shown in [13].

After the comparison of $B(5,18)$ and $B(5,19)$ groups, we can conclude some advantages and the reasons why we apply GAP in combinatorial group theory as below:

1. GAP can reduce a lot of tedious computations by hand;

2. GAP provides a large number of functions and algorithms applied in group theory;

3. GAP provides data libraries containing large classes of algebraic object, i.e. SmallGroup Library.

4. All finite groups can be presented by finitely presented groups respectively in GAP;

5. It is efficient and quick to test if a group is a $B(n, k)$ group;

6. It is accurately to find out n-element subsets, which satisfy the conditions of $B(n, k)$ group.

7. GAP provides a help manual insides the system;

8. GAP is "extensible" in that you can write your own programs in the GAP language.

Chapter 4

Some new results

4.1 Classification of $C_n \rtimes C_4$

In [9], Tan and Li proved that $C_5 \rtimes C_4 =< a, b|a^4 = b^5 = 1, b^a = b^{-1} >$ is a $B(4, 13)$ group. The lemma is as following:

Lemma 4.1.1. *Let $G =< a, b|a^4 = b^5 = 1, b^a = b^{-1} >$ Then G is a $B(4, 13)$ group.*

Next, we continue to study the group $G = C_n \rtimes C_4 =< a, b|a^4 = b^n = 1, b^a = b^{-1} >$. We used GAP program to check $C_n \rtimes C_4 =< a, b|a^4 = b^n = 1, b^a = b^{-1} >$, where n is equal to 5, 7, 9, 11, 13, 15. The result of the program show that $C_5 \rtimes C_4 =< a, b|a^4 = b^5 = 1, b^a = b^{-1} >$ is a $B(4, 13)$ group and $C_n \rtimes C_4 =< a, b|a^4 = b^5 = 1, b^a = b^{-1} >$ is a $B(4, 14)$ group, where $n = 7, 9, 11, 13, 15$. Next, We will show $C_n \rtimes C_4 =< a, b|a^4 = b^n = 1, b^a = b^{-1} >$ is a $B(4, 14)$ group, where $n > 7$. We complete the proof of the proposition as follows.

Proposition 4.1.2. *If a group $G = C_n \rtimes C_4 =< a, b|a^4 = b^n = 1, b^a = b^{-1} >$, where n is an odd number and $n \geqslant 7$, then G is a $B(4, 14)$-group. Moreover, G is not a $B(4, 13)$ group.*

Proof. Let a group $G = C_n \rtimes C_4 < a, b|a^4 = b^n = 1, b^a = b^{-1} >$. We divide into two steps to show that G is a $B(4, 14)$-group. The first step will show $|A^2| \geq 14$, where A is a particular subset in G and which is found by GAP program. The program is attached in the appendices. In the second step, we will show there are at least 2 duplicate elements in any A^2.

Part 1: Show $A^2 \geq 14$.

Let $A = a, b, ab, a^2b^2$. Then $A^2 = \{a^2, ab, a^2b, a^3b^2, ba, b^2, bab, ba^2b^2, aba, ab^2, aba^2b^2, a^2b^2a, a^2b^2b,$ $a^2b^2ab, a^2b^2a^2b^2\}$. Notice that $b^2 = b^{-1}$, we can simplify the square se A^2. Then $A^2 =$ $\{\underline{a^2}, \underline{ab}, \underline{a^2b}, \underline{a^3b^2}, \underline{ab^{-1}}, \underline{b^2}, \underline{a}, \underline{a^2b^3}, \underline{a^2b^{-1}}, \underline{ab^2}, a^2, \underline{a^3b^3}, \underline{a^3b^{-2}}, a^2b^3, \underline{a^3b^{-1}}, \underline{b^{-4}}\}$.

We display the power of b in the chart as following:

b^n		ab^n		a^2b^n		a^3b^n	
2	-4	1	-1	0	1	2	3
		0	2	3	-1	-2	-1

Thus, there exists at least one subset A in G, such that $|A^2| = 14$. Hence, G is not a $B(4, 13)$, when $n \geq 7$ and n is an odd number.

Part 2: Show $|A^2| \leq 14$.

Since a^2 is center of g, let $H =< b, a^2 >$. Then $G = H \cup aH$. There are two cosets, which are H, aH. Let $A = \{x_1, x_2, x_3, x_4\}$, where $x_i \in G$. To show $|A^2| \leq 14$, we need to find there exists at least 2 duplicates in A^2. Then we can conclude that there is no such square subset A^2, which has no more than 14 distinct elements. Next, we divide into 2 cases.

Case 1: $|A \cap H| \geqslant 3$

Assume $x_1, x_2, x_3 \in H$.Since H is a cyclic group, we can easy see that :

$$x_1x_2 = x_2x_1,\ x_3x_2 = x_2x_3$$

It is easy to see that there are at least 3 duplicates in A. Thus $|A^2| \leqslant 14$.

Case 2: $|A \cap H| \leqslant 2$

Since $|A \cap H| \leqslant 2$, so we have $A \cap aH| \geqslant 3$. We can divide into two subcases.

Subcase 2.1 $|A \cap H| = 2$

We may assume $x_1, x_2 \in H$. We notice that:

$$x_1 x_2 = x_2 x_1.$$

Since $|A \cap H| = 2$, we have $|A \cap aH| = 2$

Assume $x_3, x_4 \in aH$. Then we notice that:

$$a^2 = x_3^2 = x_4^2$$

Thus, there are at least 2 duplicates in this case. $|A^2| \leqslant 14$.

Subcase 2.2: $|A \cap H| \leqslant 1 \Rightarrow |A \cap aH| \geqslant 3$

Assume $x_1, x_2, x_3 \in H$. Then we can see that:

$$a^2 = x_1^2 = x_2^2 = x_3^2$$

Thus, there are at least 2 duplicates in this case. $|A^2| \leqslant 14$.

Therefore, for all subsets A in G, where $|A| = 14$, we can conclude that $|A^2| \leqslant 14$.

After completing the above two proof, part 1 show that at least one subset A such that $|A^2| \geq 14$ and part 2 prove every subset A such that $|A^2| \leqslant 14$. Combined those two parts, we can conclude that the maximum order $|A^2| = 14$.

Therefore, if a group $G = C_n \rtimes C_4 =< a, b | a^4 = b^n = 1, b^a = b^{-1} >$, where n is an odd number and $n > 7$, then G is a $B(4, 14)$-group. $\square$

In 2011, Moss verified that all D_{2n} are $B(5, 22)$ group with using GAP program in his undergraduate project. Now, we consider $G = C_n \rtimes C_4$. In [14], Tan mentions $G = C_n \rtimes C_4$

is not a $B(5,19)$ group, where $n \geq 11$ and n is an odd number. We use the GAP program to investigate the groups $C_n \rtimes C_4 =< a, b|a^4 = b^n = 1, b^a = b^{-1} >$, where n is equal to 11, 13, 15,17. Our main result shows that those four groups are all exact $B(5,22)$ groups. Here, we complete the proof the proposition as follows.

Proposition 4.1.3. *If a group $G = C_n \rtimes C_4 =< a, b|a^4 = b^n = 1, b^a = b^{-1} >$, where n is an odd number and $n \geqslant 11$, then G is a exact $B(5,22)$ groups, which is not $B(5,21)$ groups.*

Proof. Let a group $G = C_n \rtimes C_4 < a, b|a^4 = b^n = 1, b^a = b^{-1} >$. We divide into two steps to show that G is a $B(5,22)$-group. It is similar proof to the proposition 4.1.2

. **Part 1:** Showing $|A^2| \geqslant 22$.

Let $A = \{a, b, ab, b^{-3}, a^3b^{-3}\}$. Then $A^2 = \{a^2, ab, a^2b, ab^{-3}, b^{-3}, ba, b^2, bab, b^{-2}, ba^3b^{-3}, aba, ab^2, abab, ab^{-2}, aba^3b^{-3}, b^{-3}a, b^{-2}, b^{-3}ab, b^{-6}, b^{-3}a^3b^{-3}, a^3b^{-3}a, a^3b^{-3}b, a^3b^{-3}ab, a^3b^{-6}, a^3b^{-3}a^3b^{-3}\}$. Note that $b^a = b^{-1}$, we get $ba = ab^{-1}$ and simplify A^2. Then we get $A^2 = \{\underline{a^2}, \underline{ab}, \underline{a^2b}, \underline{ab^{-3}}, \underline{b^{-3}}, \underline{ab^{-1}}, \underline{b^2}, \underline{a}, \underline{b^{-2}}, \underline{a^3b^{-4}}, \underline{a^2b^{-1}}, \underline{ab^2}, a^2, \underline{ab^{-2}}, \underline{b^{-4}}, \underline{ab^3}, b^{-2}, \underline{ab^{-4}}, \underline{b^{-6}}, \underline{a^3}, \underline{b^3}, \underline{a^3b^{-2}}, \underline{b^4}, \underline{a^3b^{-6}}, a^2\}$.

The following chart displays the power of b(unerlined elements).

b^n		ab^n		a^2b^n		a^3b^n	
-3	2	1	-3	0	1	-4	0
-2	-4	-1	0	-1		-2	-6
-6	3	2	-2				
4		3	-4				

Thus, there exits at least one subset A in G, such that $A^2 = 22$.

Therefore,there is at least one subset A such that $|A^2| \geqslant 22$.

Part 2: Showing $A^2 \leqslant 22$

Since a^2 is center of g, let $H =< b, a^2 >$. Then $G = H \cup aH$. There are two cosets, which are H, aH. Let $A = \{x_1, x_2, x_3, x_4, x_5\}$, where $x_i \in G$. To show $|A^2| \leq 22$, we need to find there exists at least 3 duplicates for each square set A^2. Then we can conclude that there is no such square subset A^2, which has no more than 22 distinct elements. Next, we divide into 2 cases.

Case 1: $|A \cap H| \geqslant 3$

Assume $x_1, x_2, x_3 \in H$.Since H is a cyclic group, we can easy see that :

$$x_1x_2 = x_2x_1,\ x_3x_2 = x_2x_3,\ x_1x_3 = x_3x_1$$

It is easy to see that there are at least 3 duplicates in A. Thus $|A^2| \leqslant 22$.

Case 2: $|A \cap H| \leqslant 2$

Since $|A \cap H| \leqslant 2$, so we have $A \cap aH| \geqslant 3$. We can divide into two subcases.

Subcase 2.1 $|A \cap H| = 2$

We may assume $x_1, x_2 \in H$. We notice that:

$$x_1x_2 = x_2x_1.$$

Since $|A \cap H| = 2$, we have $|A \cap aH| = 3$

Assume $x_3, x_4, x_5 \in aH$. Then we notice that:

$$a^2 = x_3^2 = x_4^2 = x_5^2$$

Thus, there are at least 3 duplicates in this case. $|A^2| \leqslant 22$.

Subcase 2.2: $|A \cap H| \leqslant 1 \Rightarrow |A \cap aH| \geqslant 4$

Assume $x_1, x_2, x_3, x_4 \in H$. Then we can see that:

$$a^2 = x_1^2 = x_2^2 = x_3^2 = x_4^2$$

Thus, there are at least 3 duplicates in this case. $|A^2| \leqslant 22$.

Therefore, for all subsets A in G, where $|A| = 5$, we can conclude that $|A^2| \leqslant 22$.

After completing the above two proof, part 1 show that at least one subset A such that $|A^2| \geqslant 22$

and part 2 prove every subset A such that $|A^2| \leqslant 22$. Combined those two parts, we can conclude that the maximum order of $|A^2| = 22$.

Therefore, if a group $G = C_n \rtimes C_4 =< a, b | a^4 = b^n = 1, b^a = b^{-1} >$, where n is an odd number and $n > 10$, then G is a $B(5, 22)$-group. $\square$

4.2 Characterization of $B(6, 23)$ group

Recently, Tan and Zhong completed the characterizations of B_6 and B_7 groups in [16]. They proved that a group G is B_6 group if and only if G is either an abelian or a nonabelian trivial group, and a group G is B_7 group if and only if G is either an abelian or nonablelian trivial group. Here, we continue the studies and investigate $B(6, 23)$ groups. In this section, a complete characterization on $B(6, 23)$ group is given in Theorem 4.2.5.

Now, we prove the crucial lemma that a $B(6, 23)$ group of order $n \geq 115$ is a $B(5, 18)$ group.

Lemma 4.2.1. *Let G be a $B(6, 23)$ group of order $n > 115$. Then G is a $B(5, 18)$ group.*

Proof. Let G be a $B(6, 23)$ group of order > 115. Suppose on the contrary that G is not a $B(5, 18)$ group. There exists at least one subset $B = \{b_1, b_2, b_3, b_4, b_5\}$ of G such that $B^2 > 18$. Note that G is a $B(6, 23)$ group, then $B^2 \leqslant 23$. Let $B_l = \{b \in G | B^2 \cap bB \neq \emptyset\}$. For any $b \in B_l$, there exists $b_i, b_j, b_k \in B$ such that $bb_i = b_j b_k$. Then $B_l = \{b_j b_k b_i^{-1} | b_i, b_j, b_k \in B\}$. Notice that G is a $B(6.23)$ group, such that $B^2 \leq 23$. Then $|B_l| \leq 23 \times 5 = 115$. Since $|G| > 115 \geqslant |B_l|$, $G - B_l \neq \emptyset$. Thus, there exists an element $b \in G$ such $b \notin B_l$ that $bB \cap B^2 = \emptyset$. Let $A = B \cup \{b\}$. Then $A^2 = B^2 \cup bB \cup Bb \cup b^2$. Therefore, $|A^2| = |B^2 \cup bB \cup Bb \cup b^2| = |B^2| + |bB| + |Bb| + |b^2| \geqslant |B^2| + |bB| > 18 + 5 = 23$ since $bB \cap B^2$, which is a contradiction to the assumption that G is a $B(6, 23)$ group. Hence, the lemma is proved. $\square$

The classification of $B(5,18)$ groups is given by Huang and Li in the Theorem 2.2.9 in [13].

Lemma 4.2.2. *A group G is a $B(5,18)$ group if and only if one of the following statements holds:*

1. *G is abelian;*
2. *G is a nonabelian trivial $B(5,18)$ group;*
3. *$G \cong < a, b, | a^5 = b^4 = 1, a^b = a^{-1} >$.*

Since a group of order 20 is as known as a trivial $B(6,23)$ group. Let a group $G \cong < a, b, | a^5 = b^4 = 1, a^b = a^{-1} >$. Then G is a nonabelian trivial $B(6,23)$ group.

By Lemma 4.2.1 and lemma 4.2.2, a $B(5,18)$ group of order > 115 is abelian. Then we can obtain the following proposition.

Proposition 4.2.3. *Let G be a $B(6,23)$ group of order > 115. Then G is abelian.*

Next, we use GAP programs to test through all groups up to order 115. The functions are attached in appendices. We found that there are only two nonabelian nontrivial $B(6,23)$ groups. Then we can conclude the following proposition.

Proposition 4.2.4. *If a group G is nonablian nontrivial $B(6,23)$ group of order up to 115, then G is one of the following groups:*

1) $C_3 \rtimes Q_8 = < a, b, c | a^3 = b^4 = c^4, b^2 = c^2, ac = ca, a^b = a^{-1}, b^c = b^3 >$;

2) $C_3 \rtimes (C_4 \times C_2) = < a, b, c | a^3 = b^4 = c^2 = 1, bc = cb, a^b = a^{-1}, ac = ca >$.

By combining the above propositions, we obtain our main result of the characterization of $B(6,23)$ group as follows.

Theorem 4.2.5. *If a group G is $B(6,23)$ group, then G is one of the following groups:*

1) Abelian groups;

2) Nonabelian trivial $B(6,23)$ groups;

3) $C_3 \rtimes Q_8 = < a,b,c | a^3 = b^4 = c^4, b^2 = c^2, ac = ca, a^b = a^{-1}, b^c = b^3 >$;

4) $C_3 \rtimes (C_4 \times C_2) = < a,b,c | a^3 = b^4 = c^2 = 1, bc = cb, a^b = a^{-1}, ac = ca >$.

Bibliography

[1] Y. G. Berkovich, G. A. Freiman and C. E. Praeger, Small squaring and cubing properties for finite groups, Bull. Austral. Math. Soc. 44 (1991), 429–450.

[2] T. Eddy and M.M. Parmenter, Groups with restricted squaring properties, Ars Combin. 104 (2012) 321-331.

[3] G.A. Freiman, On two- and three-element subsets of groups, Aequationes Math. 22 (1981), 140-152.

[4] P. Longobardi and M. Maj, The classification of groups with the small squaring property on 3-sets, Bull. Aust. Math. Soc. 46 (1992), 263-269.

[5] M.M. Parmenter, On groups with redundacy in multiplication, Ars Combin. 63 (2002), 119-127.

[6] Y. Li and X. Pan, On $B(5,k)$-groups, Bull. Aust. Math. Soc. 84 (2011) 393-407.

[7] Y. Li and X. Pan, On $B(5,k)$-groups, Thesis, June 9, 2011, Brock University.

[8] Y. Li and Y. Tan, On B_5-groups, Ars Combin., 114 (2014) 3-14.

[9] Y. Li and Y. Tan, On $B(4,k)$ groups, J. Algebra Appl. 9 (2010), 27-42.

[10] Y. Li and Y. Tan, On $B(4,13)$ 2-groups. Comm. Algebra, 39 (10) (2011) 3769-3780.

[11] H. Wang, Y. Tan and T. Moss, On $B(n,k)$ 2-groups, . Comm. Algerbra 43 (2015),no. 11, 4622-4659.

[12] Y. Li and H. Huang, On $B(4,14)$ 2-groups,J. Algebra Appl. 14 (2015), no. 8, 1550118, 14 pp.

[13] Y. Li and H. Huang, On $B(5,18)$ groups, Comm. Algebra 44 (2016), no. 2, 568-590.

[14] Y.Tan and T.Zhong (2019), On $B(5,19)$ non-2-groups, Comm. Algebra, DOI: 10.1080/00927872.2019.1654494

[15] B. Huppert, Endliche Gruppen I, Springer-verlag, Berlin-Heidelberg- New York, 1967.

[16] Y.Tan and T.Zhong (2020), On B_6 and B_7-groups. Comm. Algebra 48 (2020), no. 2, 663–667. 20D60 (20A05 20E34)

[17] D.J.S Robinson, A course in the groups. Springer-Verlag, berlin-Heidelberg-New York, 1982.

[18] Rotman, J. (1995). An introduction to the theory of groups (4th ed.). New York: Springer-Verlag.

[19] The GAP Group, *GAP – Groups, Algorithms, and Programming, Version 4.10.2*; 2019, (`https://www.gap-system.org`).

Appendices

Appendix A

GAP programs

A.1 B(5,19)

```
 # "#" means comments of the program.
 # Instruction of program
 # This program is to check a group is B(5,19)-group.
 # The output of the program is the ID of a SmallGroup with the largest order k,
 # which group is B(5,k)-group.
 # To execute this function, create a file call B5kk.g on the desktop  first.
 # Use the following command to run this function.
 # Read("Desktop/B5kk.g"); g:=SmallGroup(20,1); B5kk(g);
B5k:=function(g,k)
    local m,count,max,ord,id,A,testA2,i1,i2,i3,i4,i5, OrdOfGroup;
    i1:=0;i2:=0;i3:=0;i4:=0;i5:=0;
    testA2:=[]; A:=[]; count :=0; max:=0;
    ord:=Order(g);
    m:= MultiplicationTable(g);
    for i1 in [1..ord] do
      for i2 in [i1..ord] do
          for i3 in [i2..ord] do
              for i4 in [i3..ord] do
                 for i5 in [i4..ord] do
                  A:=[i1,i2,i3,i4,i5];
                  testA2:=m{A}{A};
                  count :=Size(Unique(Concatenation(testA2)));
                  max:=Maximum(count,max);
                  if max >k then
                     OrdOfGroup:=Concatenation(["ORDER:"],[max]);
```

```
                     return OrdOfGroup;
                  fi;
                 od;
            od;
        od;
    od;
od;
OrdOfGroup:=Concatenation(["ORDER:"],[max]);
return OrdOfGroup;
end;
```

A.2 Nonabelian group

```
 # Instruction of program
 # This program is to find all nonabelian SmallGroups with order n.
 # This program is to output the list of ID of the SmallGroups.
 # (PS: the first of the list is the number of nonabelian SmallGroups)
 # To execute this function, create a file named "nonabelian.g" on desktop.
 # Use the following commands to run an example.
 # Read("Desktop/nonabelian.g"); nonabelian(24); # find all abelian group
 # with order 24.
nonabelian := function(n)   # n for the order of a group
    local j,g,list,count;   # g for the SmallGroup
    count:=0;               # count for counting how many nonabelian group
    j:=0; g:=(); list:=[];  # initailizing
    for j in [ 1 .. NrSmallGroups( n ) ] do  # check all the SmallGroup with order n
        g := SmallGroup( n, j );
        if IsAbelian( g ) = false then
            count:=count + 1;
            AddSet(list,[n,j]);
        fi;
    od;
    return list;             # output a list of ID for the SmallGroups
end;

RemoveFirstN:=function(a,n)
    local i;
    for i in [1..n] do
       Remove(a,1);
    od;
    return a;
end;
```

A.3 Find subset A

```
 # Instruction of program
 # This program is to find all the 5 suitable elements within a smallgroup
 # with order n. For all the combination of 5 elements, A^2  is a B(5,19) group.
 # To compile and execute this function, read the file into the system first.
 # Use the following command to execute the funtion. An example is shown as
 # following: Read("Desktop/findElts.g"); g:= SmallGroup(20,1); findElts(g);
findElts:= function(g)
    local m,count,max,ord,id,A,testA2,i1,i2,i3,i4,i5, num;
    i1:=0;i2:=0;i3:=0;i4:=0;i5:=0;
    testA2:=[];
    A:=[];
    count :=0;
    num:=0;
    max:=0;
    ord:=Size(g);
    m:= MultiplicationTable(g);
    for i1 in [1..ord] do
        for i2 in [i1..ord] do
            for i3 in [i2..ord] do
                for i4 in [i3..ord] do
                    for i5 in [i4..ord] do
                        max:=0;
                        A:=[i1,i2,i3,i4,i5];
                        testA2:=m{A}{A};
                        count :=Size(Unique(Concatenation(testA2)));
                        max:=Maximum(count,max);
                        count:=Size(Unique(A));
                        if max = 19 then #and count = 5 then
                            Print(A,"\n");
                            num:=num +1;
                        fi;
                        if num=2000 then
                            return num;
                        fi;
                    od;
                od;
            od;
        od;
    od;
    return num;
end;
```

A.4 $C_n \rtimes C_4$

```
# List all group structure of order 4*p, where p is equal to 5,7,9,11,13,15,17
# An example shows the use of the function.
# Create a file named "ShowStructure.g" on Desktop.
# Read("Desktop/ShowStructure.g"); ShowStructure();
ShowStructure:=function()
    local p,d,result,count;
    p:=0;
    d:=[];
    result:=[];
    for p in [5,7,9,11,13,15,17] do
       result:=Concatenation(result,[p*4,":"]);
       d:=List(AllSmallGroups(4*p),StructureDescription);
       result:=Concatenation(result,d);
       Print(d,"\n");
    od;
    return result;
end;

solvek:=function(n)
    local i,k,list;
    list:=[];
    i:=0;
    for i in [2..n] do
        if((i^4-1) mod n = 0) then
            Add(list,i);
        fi;
    od;
    return list;
end;

B4k:=function(g)
    local m,count,max,ord,id,A,testA2,i1,i2,i3,i4, OrdOfGroup;
    i1:=0;i2:=0;i3:=0;i4:=0;
    testA2:=[];
    A:=[];
    count :=0;
    max:=0;
    ord:=Size(g);
    m:= MultiplicationTable(g);
    for i1 in [1..ord] do
        for i2 in [i1..ord] do
            for i3 in [i2..ord] do
```

```
                for i4 in [i3..ord] do
                    A:=[i1,i2,i3,i4];
                    testA2:=m{A}{A};
                    count :=Size(Unique(Concatenation(testA2)));
                    max:=Maximum(count,max);
                    if max >19 then
                        OrdOfGroup:=Concatenation(["The value of k is"],[max]);
                        return OrdOfGroup;
                    fi;
                od;
            od;
        od;
    od;
    OrdOfGroup:=Concatenation(["The value of k is"],[max]);
    return OrdOfGroup;
end;

findEB4k:= function(g,k)
    local m,count,max,ord,id,A,testA2,i1,i2,i3,i4,i5, num;
    i1:=0;i2:=0;i3:=0;i4:=0;
    testA2:=[];
    A:=[];
    count :=0;
    num:=0;
    max:=0;
    ord:=Size(g);
    m:= MultiplicationTable(g);
    for i1 in [1..ord] do
        for i2 in [i1..ord] do
            for i3 in [i2..ord] do
                for i4 in [i3..ord] do
                    max:=0;
                    A:=[i1,i2,i3,i4];
                    testA2:=m{A}{A};
                    count :=Size(Unique(Concatenation(testA2)));
                    max:=Maximum(count,max);
                    count:=Size(Unique(A));
                    if max = k then #and count = 5 then
                        Print(A,"\n");
                        num:=num +1;
                    fi;
                    if num=2000 then
                        return num;
                    fi;
                od;
            od;
```

```
        od;
    od;
    return num;
end;
```

A.5 List some properties a group

```
# This functions is to show some properties of a group.
# An example show how to execute the function.
# Create a file named "GetRelatorsOfGroup.g" on Desktop.
# Read("Desktop/GetRelatorsOfGroup.g"); g:=Small Group(24,3);
# GetRelatorsOfGroup(g);
GetRelatorsOfGroup:=function(g)
    local iso, isos,image,relators,center,G,fp,H,hom,K,s;
    Print("G = ",g,", \n");
    s:=StructureDescription(g);
    Print("The structure of group G: ", s,"\n");
    iso:=IsomorphismFpGroup(g);
    image :=Image(iso);
    relators:=RelatorsOfFpGroup(image);
    Print("The group representation is ",relators,"\n");
    isos:=IsomorphismSimplifiedFpGroup(image);
    image:=Image(isos);
    Print(image,"\n");
    s:=StructureDescription(image);
    Print(s,"\n");
    Print("Center is ", Center(Image(isos)),"\n");
end;
```

A.6 List structures of some groups with small order

```
StructureofGroups:=function(list)
    local i,size,id;
    size:=Size(list);
    for i in [1..size] do
       id:=list[i];
       Print(StructureDescription(SmallGroup(id[1],id[2])),"\n");
    od;
    return ;
end;
```

```
B520_2group:=function()
    local id, list,g;
    id:=[];
    list:=[];
    id:=[32,23];    Add(list,id);
    id:=[64,194];    Add(list,id);
    id:=[128,2152];    Add(list,id);
    return list;
end;

RelatorsofGroups:=function(list)
    local i,size,id,g,iso,isoo;
    size:=Size(list);
    for i in [1..size] do
        id:=list[i];
        g:=SmallGroup(id[1],id[2]);
        iso:=IsomorphismFpGroup(g);
        isoo:=IsomorphismSimplifiedFpGroup(Image(iso));
        Print(list[i],"  ", RelatorsOfFpGroup(Image(isoo)),"\n");
    od;
    return ;
end;

B520_non2group:=function()
    local id, list,g;
    id:=[];    list:=[];
    #B(5.20)-groups
    id:=[20,3];    Add(list,id);
    id:=[20,4];    Add(list,id);
    id:=[21,1];    Add(list,id);
    id:=[22,1];    Add(list,id);
    id:=[24,1];    Add(list,id);
    id:=[24,5];    Add(list,id);
    id:=[24,6];    Add(list,id);
    id:=[24,8];    Add(list,id);
    id:=[24,10];    Add(list,id);
    id:=[24,14];    Add(list,id);

    #B(5,18)-group
    id:=[20,1];    Add(list,id);
    #B(5,19)-groups
    id:=[24,4];    Add(list,id);
    id:=[24,7];    Add(list,id);
    id:=[24,11];    Add(list,id);
    return list;
```

```
end;
```

A.7 Characterization on $B(6, 23)$ group

```
 # Instruction of program
 # This program is to check a group is B6k group.
 # The output of the program is the ID of a SmallGroup with the
 # largest order k, which group is B5k group.
B6kk:=function(g,k)
    local m,count,max,ord,id,A,testA2,i1,i2,i3,i4,i5,i6, OrdOfGroup;
    i1:=0;i2:=0;i3:=0;i4:=0;i5:=0;i6:=0;
    testA2:=[]; A:=[]; count :=0; max:=0;
    ord:=g[1];
    id:=g[2];
    m:= MultiplicationTable(SmallGroup(ord,id));
    for i1 in [1..ord] do
        for i2 in [i1..ord] do
        for i3 in [i2..ord] do
            for i4 in [i3..ord] do
                for i5 in [i4..ord] do
                    for i6 in [i5..ord] do
                        A:=[i1,i2,i3,i4,i5,i6];
                        testA2:=m{A}{A};
                        count :=Size(Unique(Concatenation(testA2)));
                        max:=Maximum(count,max);
                        if max >k then
                           OrdOfGroup:=Concatenation(g,["ORDER:",max]);
                           return OrdOfGroup;
                        fi;
                   od;
                od;
             od;
         od;
       od;
   od;
   OrdOfGroup:=Concatenation(g,["ORDER:",max]);
   return OrdOfGroup;
end;
PrintAllBnk:=function ( a )
    local i,order,k;
    i := 0;
    for i in [ 1 .. Size( a )  ] do
        Print( B6kk( a[i] ,23),"\n" );
```

```
    od;
    return 0;
end;
nonabelian := function(n)   # n for the order of a group
    local j,g,list,count;   # g for the SmallGroup
    count:=0; j:=0; g:=(); list:=[];  # initailizing
    for j in [ 1 .. NrSmallGroups( n ) ] do  # check all the SmallGroup with order n
        g := SmallGroup( n, j );
        if IsAbelian( g ) = false then
            count:=count + 1;
            #Print( IdGroup( g ), " ", StructureDescription( g ), "\n" );
            AddSet(list,[n,j]);
        fi;
    od;
    return list;             # output a list of ID for the SmallGroups
end;

RemoveFirstN:=function(a,n)
    local i;
    for i in [1..n] do
       Remove(a,1);
    od;
    return a;
end;

#This function is to test all nonabelian groups with order
# x<n<y if a group is B(5,19) group.
test:=function(x,y)
    local n,list,a;
    list:=[]; a:=[]; n:=0;
    for n in [x..y] do
        if(n<>32 and n<>64 and n<>128 and n<> 256) then
            a:=nonabelian(n);
            if Size(a) > 0 then
                PrintAllBnk(a);
            fi;
        fi;
    od;
    return 0;
end;
test2p:=function(n)
    local list,a;
    list:=[]; a:=[];
    a:=nonabelian(n);
    if Size(a) > 0 then
       list:=PrintAllBnk(a);
```

```
    fi;
    return 0;
end;
#Introduction to test B(6,23) group up to order 115:
1. Create a file named "B623.g" on desktop,
2. Open GAP software,
3. Read the file from the desktop by calling Read("Desktop/B623.g");
4. Type the function "test(24,115)".
5. Type the function "test2p(32)"; "test2p(64)".
```

A.8 Representations of nonabelian groups of order 24

```
This method is to present all 12 nonabelian groups of order 24 up to
isomorphism by using the Smallgroup Library in GAP.
The First step is list the ID of nonabelian groups of order 24.
gap> Read("Desktop/nonAb.g");
gap> nonabelian(24);
[ [ 24, 1 ], [ 24, 3 ], [ 24, 4 ], [ 24, 5 ], [ 24, 6 ], [ 24, 7 ],
 [ 24, 8 ], [ 24, 10 ], [ 24, 11 ], [ 24, 12 ], [ 24, 13 ], [ 24, 14]]
The next step is to present some nonabelian groups of 24 as below.
1. [24,1] = C3 : C8;
gap> g1:=SmallGroup(24,1);
<pc group of size 24 with 4 generators>
gap> StructureDescription(g1);
"C3 : C8"
gap> iso:=IsomorphismFpGroup(g1);
[ f1, f2, f3, f4 ] -> [ F1, F2, F3, F4 ]
gap> isoo:=IsomorphismSimplifiedFpGroup(Image(iso));
[ F1, F2, F3, F4 ] -> [ F1, F1^2, F1^-4, F4 ]
gap> image:=Image(isoo);
<fp group on the generators [ F1, F4 ]>
gap> RelatorsOfFpGroup(image);
[ F4^3, F1^-1*F4*F1*F4, F1^8 ]
2. [24,3] = SL(2, 3) special linear group
gap> g2:=SmallGroup(24,3);
<pc group of size 24 with 4 generators>
gap> StructureDescription(g2);
"SL(2,3)"
gap> iso:=IsomorphismFpGroup(g2);
[ f1, f2, f3, f4 ] -> [ F1, F2, F3, F4 ]
gap> isoo:=IsomorphismSimplifiedFpGroup(Image(iso));
[ F1, F2, F3, F4 ] -> [ F1, F2, F1^-1*F2*F1, F2^-2 ]
gap> image:=Image(isoo);
```

```
<fp group on the generators [ F1, F2 ]>
gap> RelatorsOfFpGroup(image);
[ F1^3, F2^4, F2^-1*F1*F2^2*F1^-1*F2^-1, F2*(F1*F2^-1)^2*F1 ]
gap> g3:=SmallGroup(24,4);
<pc group of size 24 with 4 generators>
3. [24,3] = C3 : Q8
gap> StructureDescription(g3);
"C3 : Q8"
gap> iso:=IsomorphismFpGroup(g3);
[ f1, f2, f3, f4 ] -> [ F1, F2, F3, F4 ]
gap> isoo:=IsomorphismSimplifiedFpGroup(Image(iso));
[ F1, F2, F3, F4 ] -> [ F1, F2, F1^-2, F4 ]
gap> image:=Image(isoo);
<fp group on the generators [ F1, F2, F4 ]>
gap> RelatorsOfFpGroup(image);
[ F4^3, F2*F1^-1*F2^-1*F1^-1, F2*F1^2*F2, F2*F1^-1*F2*F1,
F1^-1*F4*F1*F4, F4^-1*F2^-1*F4*F2 ]
4. [24, 5] = C4 X S3
gap> g4:=SmallGroup(24,5);
<pc group of size 24 with 4 generators>
gap> StructureDescription(g4);
"C4 X S3"
gap> iso:=IsomorphismFpGroup(g4);
[ f1, f2, f3, f4 ] -> [ F1, F2, F3, F4 ]
gap> isoo:=IsomorphismSimplifiedFpGroup(Image(iso));
[ F1, F2, F3, F4 ] -> [ F1, F2, F2^-2, F4 ]
gap> image:=Image(isoo);
<fp group on the generators [ F1, F2, F4 ]>
gap> RelatorsOfFpGroup(image);
[ F1^2, F4^3, F2^4, (F1*F4)^2, F2^-1*F1*F2*F1, F4^-1*F2^-1*F4*F2 ]
5. [24, 6] = D24
gap> g5:=SmallGroup(24,6);
<pc group of size 24 with 4 generators>
gap> StructureDescription(g5);
"D24"
gap> iso:=IsomorphismFpGroup(g5);
[ f1, f2, f3, f4 ] -> [ F1, F2, F3, F4 ]
gap> isoo:=IsomorphismSimplifiedFpGroup(Image(iso));
[ F1, F2, F3, F4 ] -> [ F1, F2, F2^-2, F4 ]
gap> RelatorsOfFpGroup(image);
[ F1^2, F4^3, F2^4, (F1*F4)^2, F2^-1*F1*F2*F1, F4^-1*F2^-1*F4*F2 ]
6. [24, 7] = C2 X (C3 : C4)
gap> g6:=SmallGroup(24,7);
<pc group of size 24 with 4 generators>
gap> StructureDescription(g6);
"C2 X (C3 : C4)"
```

```
gap> iso:=IsomorphismFpGroup(g6);
[ f1, f2, f3, f4 ] -> [ F1, F2, F3, F4 ]
gap> isoo:=IsomorphismSimplifiedFpGroup(Image(iso));
[ F1, F2, F3, F4 ] -> [ F1, F2, F1^-2, F4 ]
gap> image:=Image(isoo);
<fp group on the generators [ F1, F2, F4 ]>
gap> RelatorsOfFpGroup(image);
[ F2^2, F4^3, F1^4, F1^-1*F4*F1*F4, F2*F1^-1*F2*F1, F4^-1*F2*F4*F2 ]
7. [24, 8] = (C6 x C2) : C2
gap> g7:=SmallGroup(24,8);
<pc group of size 24 with 4 generators>
gap> StructureDescription(g7);
"(C6 X C2) : C2"
gap> iso:=IsomorphismFpGroup(g7);
[ f1, f2, f3, f4 ] -> [ F1, F2, F3, F4 ]
gap> isoo:=IsomorphismSimplifiedFpGroup(Image(iso));
[ F1, F2, F3, F4 ] -> [ F1, F2, (F2*F1)^2, F4 ]
gap> image:=Image(isoo);
<fp group on the generators [ F1, F2, F4 ]>
gap> RelatorsOfFpGroup(image);
[ F1^2, F2^2, F4^3, (F1*F4)^2, F4^-1*F2*F4*F2, (F2*F1)^4 ]
8. [24, 10] = C3 X D8
gap> g8:=SmallGroup(24,10);
<pc group of size 24 with 4 generators>
gap> StructureDescription(g8);
"C3 X D8"
gap> iso:=IsomorphismFpGroup(g8);
[ f1, f2, f3, f4 ] -> [ F1, F2, F3, F4 ]
gap> isoo:=IsomorphismSimplifiedFpGroup(Image(iso));
[ F1, F2, F3, F4 ] -> [ F1, F2, F3, (F2*F1)^2 ]
gap> image:=Image(isoo);
<fp group on the generators [ F1, F2, F3 ]>
gap> RelatorsOfFpGroup(image);
[ F1^2, F2^2, F3^3, F3^-1*F1*F3*F1, F3^-1*F2*F3*F2, (F2*F1)^4 ]
9. [24, 11] = C3 X Q8
gap> g9:=SmallGroup(24,11);
<pc group of size 24 with 4 generators>
gap> StructureDescription(g9);
"C3 X Q8"
gap> iso:=IsomorphismFpGroup(g9);
[ f1, f2, f3, f4 ] -> [ F1, F2, F3, F4 ]
gap> isoo:=IsomorphismSimplifiedFpGroup(Image(iso));
[ F1, F2, F3, F4 ] -> [ F1, F2, F3, F1^-2 ]
gap> image:=Image(isoo);
<fp group on the generators [ F1, F2, F3 ]>
gap> RelatorsOfFpGroup(image);
```

```
[ F3^3, F2*F1^-1*F2*F1, F1^-1*F2^-2*F1^-1, F1*F2^-1*F1*F2,
F3^-1*F1^-1*F3*F1, F3^-1*F2^-1*F3*F2 ]
10. [24, 12] = S4
gap> g10:=SmallGroup(24,12);
<pc group of size 24 with 4 generators>
gap> StructureDescription(g10);
"S4"
gap> iso:=IsomorphismFpGroup(g10);
[ f1, f2, f3, f4 ] -> [ F1, F2, F3, F4 ]
gap> isoo:=IsomorphismSimplifiedFpGroup(Image(iso));
[ F1, F2, F3, F4 ] -> [ F1, F2, F3, F1*F3*F1 ]
gap> image:=Image(isoo);
<fp group on the generators [ F1, F2, F3 ]>
gap> RelatorsOfFpGroup(image);
[ F1^2, F3^2, F2^3, (F1*F2)^2, F2^-1*F3*F2*F1*F3*F1, (F2^-1*F3)^3 ]
11. [24, 13] = C2 X A4
gap> StructureDescription(g11);
"C2 X A4"
gap> iso:=IsomorphismFpGroup(g11);
[ f1, f2, f3, f4 ] -> [ F1, F2, F3, F4 ]
gap> isoo:=IsomorphismSimplifiedFpGroup(Image(iso));
[ F1, F2, F3, F4 ] -> [ F1, F2, F3, F2^-1*F3*F2 ]
gap> image:=Image(isoo);
<fp group on the generators [ F1, F2, F3 ]>
gap> RelatorsOfFpGroup(image);
[ F1^2, F3^2, F2^3, F2^-1*F1*F2*F1, (F3*F1)^2, (F3*F2)^3 ]
12. [24, 14] = C2 X C2 X S3
gap> g12:=SmallGroup(24,14);
<pc group of size 24 with 4 generators>
gap> StructureDescription(g12);
"C2 X C2 X S3"
gap> iso:=IsomorphismFpGroup(g12);
[ f1, f2, f3, f4 ] -> [ F1, F2, F3, F4 ]
gap> isoo:=IsomorphismSimplifiedFpGroup(Image(iso));
[ F1, F2, F3, F4 ] -> [ F1, F2, F3, F4 ]
gap> image:=Image(isoo);
<fp group on the generators [ F1, F2, F3, F4 ]>
gap> RelatorsOfFpGroup(image);
[ F1^2, F2^2, F3^2, F4^3, (F1*F4)^2, (F2*F1)^2, (F3*F1)^2,
(F3*F2)^2, F4^-1*F2*F4*F2, F4^-1*F3*F4*F3 ]
```

This method is used the finitely presented group to present all the nonabelian groups of 24. In this method, we create a free group with the specific number of generators. This example will show some of the nonabelian groups as following.

```
  g2 = SL(2,3)
gap> f:=FreeGroup("a","b");
<free group on the generators [ a, b ]>
gap> g2:=f/[f.1^3,f.2^4,f.2^-1*f.1*f.2^2*f.1^-1*f.2^-1,
f.2*(f.1*f.2^-1)^2*f.1];
<fp group on the generators [ a, b ]>
gap> StructureDescription(g2);
"SL(2,3)"
gap> RelatorsOfFpGroup(g2);
[ a^3, b^4, b^-1*a*b^2*a^-1*b^-1, b*(a*b^-1)^2*a ]
gap> f:=FreeGroup("a","b","c");
<free group on the generators [ a, b, c ]>
gap> g2:=f/[f.1^3*f.2^-3,f.1^3*f.3^-2,f.3^-1*f.2^-1*f.1^2];
<fp group on the generators [ a, b, c ]>
gap> StructureDescription(g2);
"SL(2,3)"
gap> RelatorsOfFpGroup(g2);
[ a^3*b^-3, a^3*c^-2, c^-1*b^-1*a^2 ]
  g3 = C3 : Q8
gap> f:=FreeGroup("a","b","c");
<free group on the generators [ a, b, c ]>
gap> g3:=f/[f.1^3,f.2^4,f.2^2*f.3^-2, f.1^-1*f.3^-1*f.1*f.3,
f.2^-1*f.1*f.2*f.1,f.3^-1*f.2*f.3*f.2];
<fp group on the generators [ a, b, c ]>
gap> StructureDescription(g3);
"C3 : Q8"
gap> RelatorsOfFpGroup(g3);
[ a^3, b^4, b^2*c^-2, a^-1*c^-1*a*c, b^-1*a*b*a, c^-1*b*c*b ]
  g5 = D24
gap> f:=FreeGroup("a","b");
<free group on the generators [ a, b ]>
gap> g5:=f/[f.1^12,f.2^2, f.2^-1*f.1*f.2*f.1];
<fp group on the generators [ a, b ]>
gap> StructureDescription(g5);
"D24"
gap> RelatorsOfFpGroup(g5);
[ a^12, b^2, b^-1*a*b*a ]
  g10 = S4
gap> f:=FreeGroup("a","b","c");
<free group on the generators [ a, b, c ]>
gap> g10:=f/[f.1^2,f.2^3,f.3^4, f.1*f.2*f.3];
<fp group on the generators [ a, b, c ]>
gap> StructureDescription(g10);
"S4"
gap> RelatorsOfFpGroup(g10);
[ a^2, b^3, c^4, a*b*c ]
```