Finite Group Behavior, a Software Package For Beginning Group Theory

Edward Keppelmann Bayard Webb University of Nevada

June 2, 2004

Abstract

In 1989 Ladnor Geissinger at the University of North Carolina Chapel Hill developed an extraordinary DOS program called Exploring Small Groups (ESG) for students learning beginning group theory. In 1996, Ellen Maycock Parker from Depauw University augmented ESG with a series of labs and instructional materials. In this paper we announce the advent of a windows based program called Finite Group Behavior (FGB) which improves on the capabilities and instructional power of ESG. The program is available free of charge from our website. In this article we discuss the features and philosophy of FGB and we offer some suggestions for its use. In conjunction with Parker's labs, our perspective will give the instructor of beginning abstract algebra some very good ideas about using FGB to convey the intricacies of group theory as well as the sense of discovery which any research mathematician knows intimately.

1 Introduction

As an undergraduate during the fall of 1994, at the University of Nevada Reno, Bayard Webb was a student of Ed Keppelmann in Math 331: Groups, Rings, and Fields. For the portion of the course devoted to group theory, the main concepts were motivated by a computer component based on the DOS software *Exploring Small Groups*, by Ladnor Geissinger [4] and labs written by Ellen Maycock Parker [7].[1] With a blend of interests as a highly motivated and curious undergraduate mathematics major and a talented computer programmer, Webb took it upon himself to rewrite and expand on the capabilities of ESG in a windows-based environment.

Unlike advanced computational tools such as MAGMA [5] or GAP [3], FGB is intended as a teaching tool for those at the beginning level of group theory. One of the first steps in the study of groups can be the study of concrete examples, in the form of their Cayley tables, along with just enough functionality so that subgroups, cosets, and homomorphisms can be easily constructed. Of course more than the finite groups represented here should be studied in such a course, but this is an important start where conceptual understanding can pave the way for other investigations. Too much computational power can make answers

[1] The reference [4] is provided for completeness but is no longer being published. [7] contains the software of [4].

come in a black box fashion, thereby short circuiting the learning process. In contrast, what FGB does can (and often should) be replicated by the beginning student with pencil and paper on a much smaller scale.

The results of Geissinger's original and profound conception of ESG as a learning tool, along with Webb's persistence and talent, are now available at no charge from our website (see the appendix for the address and installation instructions). After a brief registration questionnaire, anyone is welcome to use the software free of charge. As well as keeping future users informed about upgrades, the registration information will be used as evidence of need and interest to request support in the form of grant applications, faculty release time and other forms of recognition for future developments. Suggestions, criticisms, and ideas for improvements are always encouraged.

2 Learning objectives

Abstract algebra is arguably one of the most difficult courses in an undergraduate mathematics curriculum. Traditionally, the abstract nature of the course, coupled with a lack of accessible examples, created a situation in which very few students were able to grasp and appreciate the subject at a suitable level. As in a beginning linear algebra course (but perhaps even more difficult here), it is important to teach the abstract nature of the course while still finding the time to suitably motivate the material with meaningful examples. Without a variety of such examples, the student can easily be left feeling that he/she has just acquired a good bit of mostly useless abstract nonsense.

Because the software is really just a computational tool, and not a collection of pre-organized lessons, it possesses a vast variety of potential uses, and we sincerely hope that our work will lead to new and improved lab manuals like those in [4]. In this paper we will mostly content ourselves with a discussion of the power of the software and an overview of its basic uses. However, in order to illustrate the possibilities, we have also included a presentation of some rudimentary examples for learning with the program. We are always eager to consider building in new functionality into FGB by adding features which the academic community thinks are necessary. We will close the paper with some of the additions we are currently considering.

In specific terms, we believe that FGB is very well suited for teaching the following concepts and objectives:

- The axioms of a group and how these are verified from a Cayley table for the case of a finite group.
- Basic notions in a group such as subgroups and orders of elements.
- Group isomorphisms (especially how to recognize when they will exist and how to construct them).
- Cosets, normal subgroups, and factor groups (easily the hardest concepts for even the best students).

- Homomorphisms and the fundamental isomorphism theorem (i.e., for $\varphi : G \to H$ we have that $G/\ker\varphi \cong \varphi(G)$).
- Learning to reason efficiently about such problems as finding the complete subgroup lattice for a group and constructing homomorphisms or finding groups and subgroups with prescribed properties.
- Forming conjectures about finite group properties to eventually either prove or construct counterexamples.

Before considering using FGB for homework or lab assignments, an important note of caution is the following. Assignments must be chosen with a lot of thought and presented with proper guidance. Some of the more popular ESG or FGB labs ask students to perform tasks such as finding the subgroup lattice for a given group or constructing homomorphisms with a specified kernel or image. We have found that in the presence of limited knowledge, even with the software, such tasks can appear to be an enormous amount of busy work. It is extremely important to emphasize that the more theorems and careful reasoning that a student brings to bear on problems such as these, the less busy work there is. The instructor must take great care in knowing what theorems and techniques they wish to introduce to the students when starting a lab and they must monitor student progress as the labs are completed. Each lab should then be followed with a discussion about what was discovered and what can be proved for use in the future. In particular, we always make it a point to demonstrate efficient solutions to problems at carefully chosen points within the course. The beauty of the software is that it can be used simultaneously to demonstrate theorems and to encourage the discovery and proof of more theorems.

3 An FGB Primer

Those familiar with programs from a windows environment and with the basics of group theory should find FGB self explanatory and easy to operate. However, since the beginning student will not have this expertise in group theory and perhaps also not a lot of familiarity with windows, they may need to rely on help from their instructor. In order to facilitate this process, we will use this opportunity to briefly outline the features of the program as well as to indicate some functionality that might not be so apparent.

Thanks to the efficient work of Magma [5], FGB comes equipped with a library that contains all the finite groups up to order 16, all nonabelian groups up to order 40, and also the alternating group A_5 of even permutations on 5 symbols. The files which form the groups within this library are organized by folders mostly according to their size. The group files are text files in a special format which can be read by FGB. The names of the files are given as 4 digit numbers. The first two digits are the size of the group and the second two are an ordering of the groups with this particular size. Cyclic subgroups (for sizes 16 or less) can be found in the cyclic folder and always end in 01. For all groups of size 16 or less the numbering is identical to that used by Geissinger in ESG [4]. From our website (see the appendix) you can download a single file to install FGB either on a hard drive or diskette. Installation on a diskette can be useful for students who may not have their own computer but could need to have their own version of the library which they can alter or augment (especially in the notes tab - see below) as their work requires.

Along with the classification theorem for finite Abelian groups, the completeness of this library allows the user to ask and definitively answer questions about the existence of certain features within small groups. (The center button under the commutativity tab can be used to quickly decide if a group is abelian or not). For example, with a quick examination of 2001, 2002, and 2003 (the only groups of order 20 in the library) it is immediate that all nonabelian groups of order 20 have centers of order 2 or less. As any student of beginning group theory will quickly realize, studying groups of various orders often involves studying what kinds of subgroups and factor groups can be formed from a given group. As we will shortly explain, the software allows the user to extract subgroups and factor groups (when appropriate) and easily create new group files for later analysis. The completeness of the group library means that an instructor can always be confident in asking students to identify any group which arises in this way (i.e. specify the group in the library or give an abelian direct sum decomposition which represents the isomorphism type).

Each group to be considered is opened as a separate window within FGB. When many groups are opened simultaneously these can be arranged in a tiled format or on top of each other. Each group in the library can be viewed in a form where the elements are abstract symbols as in ESG (called standard names) or in a form (called descriptive elements) where the elements are named according to their functionality. For smaller groups (of approximately size 16 or less) this descriptive format can involve a variety of possibilities including a type of direct sum notation, integers for modulo arithmetic, or permutations as in the case of S_3. The descriptive names for larger groups are given in terms of their generators and relations. (Always consult the notes tab for these details in any particular group.)

Drag and edit features give the user the ability to reorder the elements in a group or rename them in a more suitable form. You can also enter your own groups with the program. You might want to do this, for example, if you need an Abelian group of order more than 16 or if you have some abstract group that you need to identify within the library.

When a group table is opened, in addition to the usual menu tabs of File, Edit, Window and Help, (which reside within the shell window of FGB and thus not with any particular group) the following tabs are available: Axiom Check, Group Table, Subgroup Table, Commutativity, Homomorphisms, and Notes. The function of each of these is briefly outlined below:

FILE, EDIT, WINDOW, HELP provide the usual windows functions with some features specific to group theory. The **FILE** tab allows you to open or create new group file windows. The usual self explanatory features of SAVE, SAVE AS (for the group files), and EXIT are also available. The **EDIT** tab allows you to modify existing tables by adding, removing, or renaming elements. This tab works on whatever window (within the many that may be open at any time) is considered currently active. The **WINDOW** tab (or just clicking on the desired window) are the two methods of changing the window which is currently active. The **WINDOW** tab also has a command to tile all the open windows in an attempt to see as much as possible of all of them simultaneously. This can be particularly useful for in class demonstrations showing the differences in groups of small order (like the groups of order 4 or 8). At the time of this writing the **HELP** tab can only be used to remind the user to register his/her use of the software and provide authoring information.

Axiom Check provides a way to see that the Cayley tables are indeed those of groups. Tabs for checking that there is an identity and then inverses for all elements as well as an associativity check are available. When these operations fail the user is given feedback as to where the failure occurred. This provides the student with the opportunity to realize how special the structure of a group is and just how incredible it is that there are so many of these (with impressive variety) of a modest size. [2] The inverse check can also be run on a group just to produce a list of elements and their inverses for later reference.

Group Table provides a way to examine the Cayley table for a group. In addition to the functions available under the **EDIT** tab described above, it is here that the group table can be reordered, relabeled or altered on an entry by entry basis. Among other advantages, these features allow instructors to construct modified Cayley tables for use on exams and quizzes as well as to enable students to see how sensitive the axioms can be to changes in structure as small as the altering of a single entry within the table.

Subgroup Table can be used to pick and choose which elements from a group to use to form a subgroup. The program then automatically adds what is necessary in order to form closure with these choices. The size of the subgroups formed is always indicated in the upper left corner of the table for the subgroup. This means that when cyclic subgroups are formed, the order of elements can be easily identified. Subgroups can always be conjugated by any element from the original group. Once a subgroup is formed, the group can be exported to its own file window and later saved and/or modified to be used in the future. You can also view the cosets of a subgroup. This feature then reorders the table so that elements in the same coset appear contiguously. The program employs a coloring scheme (which works best when there are no more than 16 cosets) whereby each coset is assigned a unique color. As originally conceived by Geissinger in ESG [7], when the subgroup is normal this coloring scheme will indicate that the induced operation on cosets is well defined. In this case it then becomes possible to export the quotient group for further analysis in its own window. (The user should note that when exporting a coset table, the entries are replaced by a consistent representative of each coset, thus producing a Cayley table like the others in the library.) When subgroups are not normal, the conjugation option helps remind students that conjugates of nonnormal subgroups also yield nonnormal subgroups. Of course when they are normal, conjugation only acts to (possibly) reorder the elements.

Commutativity can be used to calculate the center and centralizers of the elements in a group. When nontrivial, the center is always at least one interesting example of a normal subgroup and hence a uniformly colored coset table and resulting factor group.[3]

[2]The original DOS program of ESG came with a library of non-group Cayley tables available for study. While this is not currently included with this version of FGB, it would not be hard to incorporate if there is suitable demand.

[3]In FGB 2.0 if any of these groups are desired for export or factor group formation, then you must

Homomorphism is where homomorphisms can be constructed between any two open group windows. To do this you should operate in the window that you wish to use as the domain and then select from this the window you wish to use as the range. The homomorphisms are specified in pieces in the sense that after each new definition of the function on an element from the domain the program can either go step by step through the process of indicating what this implies about the function on all the elements in the subgroup generated by your definitions or it can do this more quickly once the student understands how this works. Of course in many cases the initial specifications given will lead to a contradiction and this is indicated by the software. When this happens, it is possible, without starting completely over, to back up to the previous stage of the definition where no contradiction existed. A table format is used to display the homomorphism. This is very much like (but with different and repeat entries) the original Cayley table for the group or a subgroup except that the header columns and rows are two squares thick to indicate both the domain and image element for the homomorphism. The body of the table then shows the multiplications of image elements. This reminds the student about the group nature of the image as well as allowing them to see consequences such as the fact that every element in the image has the same sized preimage.

Notes contains background information about the group (such as the common name for the group, format for the descriptive elements or generators and relations) as well as providing you a place to make notes about your investigations of the group.

As we have previously mentioned, for groups over size 16, the presentation and other group information was provided by Magma [5]. In this, Magma uses $x \wedge y$ to denote $y^{-1}xy$ and (x, y) to denote $x^{-1}y^{-1}xy$. A description of some groups also appears when the group has a common identification such as dihedral, symmetric, alternating, or can be written as the direct sum of smaller (possibly involving Abelian or even cyclic terms) groups. In these identifications, Dihedral(n) is used to denote the group of symmetries of the regular polygon with n sides. Occasionally, such products may also involve notation such as GroupOfOrder(12,3). This is Magma's own internal notation for some group of order 12. (A good exercise might be to determine, with justification, exactly which group of order 12 in the library this is and explain why). Since newer versions of Magma no longer provide this identification information (the presentation is only available now), it is no longer easily possible to make effective systematic use of such notation.

When working with the **Group Table** tab, clicking with the right mouse button gives you the option of copying the entire table to the notes page for printing later. These tables can then be pasted from the notes to any spreadsheet where it is possible to format and print them as you wish for use on exams and quizzes. We have had great success with simple questions on exams and quizzes which force the student to mimic (on a small scale) the functions of the software and their reasoning with the program.

When looking for subgroups with the **Subgroup Table** tab, a right click gives you

manually recreate them with the subgroup tab. This is not usually difficult since ordinarily a small subset of the elements in the group will generate the entire subgroup.

the option of copying the list of elements in the subgroup to the notes. When this happens, they will be listed in alphabetical order. This has the advantage of telling you at a glance whether subgroups produced from different generators within a given group are equal or not.

4 Example Problems

In order to give an idea as to how learning with FGB might take place, we now offer some examples for using the software. This is by no means meant to be an exhaustive study of the possibilities but rather to simply stimulate your thinking about how the concepts of beginning group theory can be addressed by this program. While we feel that much more is now possible with FGB than the old ESG, the genesis for many of these ideas is certainly due in big part to Geissinger's original work [4] and the materials of Parker [7].

In what follows, the solutions are idealized in the sense that they convey the maximum information that we would like the student to retain from the problem. Without proper guidance and follow up, many students will glean far less by much more inefficient means.

4.1 EXAMPLE 1 - FINDING SUBGROUPS

PROBLEM: *Up to isomorphism, how many subgroups of order 8 does the group* $2411 =< a, b, c, x | a^2 = b, b^2 = c, c^2 = 1, x^3 = 1, x^a = x^2 >$ *have? What are these isomorphism types?*

SOLUTION #1: Examination of the library for groups of order 8 indicates that there are 5 different groups of order 8. A brief study of these reveals the following: (Note that from any Cayley table for a group you can count the number of elements of order two as the number of times, excluding the identity, where the identity appears on the diagonal.)

0801 Cyclic $\mathbf{Z}_8$ - 1 element of order 2 and 2 elements of order 4.

0802 $\mathbf{Z}_4 \oplus \mathbf{Z}_2$ - 3 elements of order 2.

0803 $\mathbf{Z}_2 \oplus \mathbf{Z}_2 \oplus \mathbf{Z}_2$ - 7 elements of order 2.

0804 The group of symmetries of the square - 5 elements of order 2.

0805 The Quaternions - 1 element of order 2 and 6 elements of order 4.

By checking the diagonal of 2411 we see that c is the only element of order 2. Therefore, 2411 cannot have subgroups isomorphic to any of 0802, 0803, or 0804. By either using the **Subgroup table** tab to check orders or by searching on the diagonal for occurrences of c, it is not hard to see that 2411 has only two elements of order 4. This means that no subgroup of type 0805 is possible and hence the only possible subgroups of order 8 for 2411 are cyclic. by using the **Subgroup Table** to examine the subgroups produced from single elements, it is readily apparent that such cyclic subgroups do indeed exist so the answer to the original question is 1.
SOLUTION #2: From the Sylow theorems, we know that since 2411 has size 24 it will contain a subgroup of size 8 and all subgroups of size 8 will be conjugate (and hence isomorphic) to one

another. Thus we immediately know that the answer to the question is 1 and we can quickly see from the **Subgroup Table** tab that such subgroups are cyclic.

<u>COMMENTS:</u> It is our experience that a first semester course would probably not cover the Sylow theorems but they are certainly a possibility for subsequent courses as well as a check for the instructor that they have the correct solution. Despite this, it is easy to find examples of this type of problem (e.g. subgroups of size 8 in groups of order 16 or 32) where the only line of reasoning available will be more like that in SOLUTION # 1 and the final answer could be different than 1. This affords many versions of this type of problem since the library has 14 groups of order 16 and 44 groups of order 32. In the general setting, the solution to a problem like this can become quite complicated in forcing students to specifically attempt to construct the various subgroup types of order 8 within the larger group. Knowing that elements of order 4 come in pairs (i.e., x has order 4 iff x^{-1} has order 4) or that some of the groups of order 8 (i.e., 0802 and 0804) have varying numbers of noncyclic subgroups of order 4, are just some examples of the kinds of observations that can prove to be very useful in deciding which combinations of elements need to be tried (with the **Subgroup Table** tab) to produce the various groups of order 8.

4.2 EXAMPLE 2 - FACTOR GROUPS

<u>PROBLEM:</u> *Prove or disprove the following conjecture. Suppose that H and K are normal subgroups of a finite group G. If H and K are isomorphic, then G/K and G/H are isomorphic.*

<u>SOLUTION:</u> The conjecture is false and can be illustrated by many counterexamples. In addition to realizing that they cannot construct a proof, the problem for the student is to locate such an example. Part of the frustration here will no doubt be that most subgroups of the groups they will want to consider won't be normal and therefore won't yield quotient groups. One clever approach to circumvent this difficulty is to consider groups with large centers. If G is a nonabelian group with center H, then any subgroup K of H will be normal in G and will thus always yield a quotient group G/K. Consider, for example the group $3201 = < a, b, c, x, y | a^2 = x, b^2 = y, b^a = bc >$. The center of 3201 is a subgroup of order 8 generated by c, x, and y, each having order 2. Thus $\{1, x\}$ and $\{1, c\}$ are isomorphic, but $3201/\{1, x\}$ and $3201/\{1, c\}$ are not. We know this since these two groups of order 16 have different numbers of elements of order 2. $3201/\{1, x\}$ has 7 elements of order 2 while $3201/\{1, c\}$ has only 3.

4.3 EXAMPLE 3 - HOMOMORPHISMS

<u>COMMENT:</u> Problems like the following force the student to grapple with the fundamental isomorphism theorem which says that for $\varphi : G \to H$ then $G/\ker(\varphi) \cong \varphi(G)$. We encourage instructors to illustrate this theorem in class as follows: The program allows you to construct homomorphisms between any two groups. You can then use the subgroup table tab to create the factor group of the domain modulo the kernel. (You get confirmation here that the kernel is indeed normal.) This factor group can be displayed along with the image group (which will appear with the **homomorphism** tab). For reasonably sized examples, these two windows

shown side by side make the isomorphism apparent. However, you can explicitly construct the isomorphism by first properly exporting the factor group and, if φ is not onto, the image subgroup of H. The **homomorphism** tab on the exported factor group window is then the place to formulate the isomorphism, which, as you should point out, is cannonical to define.

PROBLEM: *Construct a homomorphism from* $G = 2404 = <a, b, c, x | a^2 = b^2 = x,\ x^2 = c^3 = 1,\ b^a = bx>$ *onto a group of order 4. Does it matter which group of order 4 you use? Explain.*

SOLUTION: There are exactly two groups of order 6 and two groups of order 4:

0601	$\mathbf{Z}_6$ Cyclic	0401	$\mathbf{Z}_4$ Cyclic
0602	The symmetries of an equilateral triangle	0402	$\mathbf{Z}_2 \oplus \mathbf{Z}_2$

Suppose that $\varphi : G \to H$. Since $|H| = 4$, and φ is onto, the fundamental homomorphism theorem tells us that we must have that $|\ker(\varphi)| = 6$.

Since 0602 has 3 elements of order 2 but G only has one element of order 2 (i.e., x), G cannot contain any subgroups isomorphic to 0602. Checking the orders of the elements in G reveals that there is exactly one cyclic subgroup of order 6 (i.e., that generated by cx). Since this is the only such subgroup, it is fixed by inner automorphisms and hence must be normal. Therefore, the image of φ must be isomorphic to $G/<cx>$. Looking at the coset table for this factor group we see that it has 3 elements of order two and thus must be isomorphic to 0402. **This means that there is no such homomorphism onto 0401 so it does indeed matter which group of order 4 is chosen.**

The software then shows that the homomorphism can be specified by letting $\varphi(cx) = 00$, $\varphi(b) = 10$, and $\varphi(ax) = 01$. The secret to finding a correct specification is that of course we let the kernel be $< cx >$ and then we note that elements from distinct cosets of $<cx>$ must go to distinct (nonidentity) elements in 0402. In fact any choice where b and ax go to distinct elements of order 2 in 0402 will work.

4.4 EXAMPLE 4 - THE SYLOW THEOREMS

COMMENT: The software can be used in many ways to teach the Sylow theorems. As in example 1, a knowledge of these theorems can be useful in solving many subgroup or isomorphism type questions for the groups in the library. Alternatively, after learning the statements of the theorems (and perhaps their proofs), the software can be used to illustrate their conclusions. Here we give yet a third possibility of presenting a series of problems whose solutions force the student to examine concrete examples of the concepts involved in the Sylow theorems and to construct simplified proofs for special cases of those theorems (see for example the presentations given in [2]). This should ease the way toward a more general proof. We believe that this method of teaching a proof via simplified examples will no doubt have other useful realizations with this program.

PROBLEMS: *Consider the group 2001* $G = D_{10} =< a, b, c | a^2 = b^2 = c^5 = 1, c^a = c^4 >$. *(The group of symmetries of a regular nonagon). Use G to answer the following questions:*

(a) PROBLEM: *Partition G into conjugacy classes and write the class equation for G.*

COMMENTS: Recall that for $a \in G$, the centralizer $C(a)$ of a (which can be calculated with the **Commutativity** tab), is the subgroup of all elements in G which commute with a. It is a standard fact that the number of elements in the conjugacy class of a is the index $[G : C(a)]$ of $C(a)$ in G. The class equation is then just a way of accounting for every conjugacy class in G by the formulation $|G| = \sum_i [G : C(a_i)]$ where the sum involves exactly one element from each conjugacy class. These results and notions could easily be presented in class or homework before questions like these are investigated.

This problem is also an example where the power of the software has been suitably limited to force the beginning student to think more and push buttons less. Since the program only has methods for conjugating subgroups and not single elements, the student must use ingenuity to do the calculations efficiently.

SOLUTION: Since conjugate elements have the same order we first partition G into subsets corresponding to the orders of the elements. This produces the following partition: $\{1\}$ (order 1), $\{a, b, ab, ac, ac^2, ac^3, ac^4, abc, abc^2, abc^3, abc^4\}$ (order 2), $\{c, c^2, c^3, c^4\}$ (order 5), and $\{bc, bc^2, bc^3, bc^4\}$ (order 10).

From the facts cited above about $C(x)$ we know that the size of each conjugacy class must divide $|G| = 20$. Of course $\{1\}$ is a conjugacy class and the fact that the center of G consists of $\{1, b\}$ immediately tells us that $\{b\}$ is the only other singleton conjugacy class. For the elements of order 10 we note that these must divide into two classes of size 2 since, for example, $|C(bc)| = 10$. Choosing an element, say a, not belonging to $C(bc)$ and computing $a * bc * a^{-1} = a * bc * a = abc * a = bc^4$ shows us that $\{bc, bc^4\}$ and $\{bc^2, bc^3\}$ are these two conjugacy classes. Similarly, $|C(c)| = 10$ and so with $a \notin C(c)$ we have that $a * c * a^{-1} = c^4$ so $\{c, c^4\}$ and $\{c^2, c^3\}$ give the two conjugacy classes for elements of order 5.

To compute the conjugacy classes of the elements of order 2 we can reason as follows. Since $|C(a)| = 4$, the conjugacy class of a has size 5. Furthermore, conjugating a in turn by abc, abc^3, abc^4, bc^3 (one element from each coset of $C(a)$) lists for us that the conjugacy class of a is $\{a, ac^2, ac, ac^3, ac^4\}$. Having already accounted for b above, dimension considerations immediately tell us that $\{ab, abc, abc^2, abc^3, abc^4\}$ is the remaining conjugacy class. Therefore the class equation becomes $20 = 1 + 1 + 5 + 5 + 2 + 2 + 2 + 2$. [4]

(b) PROBLEM: *Suppose that G is any group (Abelian or nonabelian) with the above class equation. Must such a group have a subgroup of order 5? Explain.*

SOLUTION: YES! Each term on the right hand side of the class equation is the index of a centralizer in G so that if we divide $|G| = 20$ by these numbers then we get the order of the centralizers, which are subgroups of G. Thus, since there are conjugacy classes of size 2 there must be centralizers of size 10. A check of the groups of size 10 (there are only 2: 1001 - $\mathbf{Z}_{10}$ and 1002 - the symmetries of a regular pentagon) shows that every group of size 10 always contains a subgroup of size 5.

[4] When computing various conjugates of a it is helpful to be able to return quickly to the subgroup $\{1, a\}$ after each conjugation. You can do this by conjugating by the inverse of any element used in the process. The **Axiom Check** tab can be used to provide a list of elements and their inverses for easy reference.

(c) PROBLEM: *In the group 2001 how many subgroups of order 4 are there and what isomorphism types occur?*

SOLUTION: The check of orders made in part (a) shows that G has no elements of order 4 and thus the only possible subgroups of order 4 are of type 0402 ($\mathbf{Z}_2 \oplus \mathbf{Z}_2$). Such a subgroup will be formed from 3 elements of order 2. Since 2001 has eleven elements of order two we do not want to attempt to find all such subgroups simply by trial and error. However, it is reasonable to use trial and error to find one of these, say $H = \{1, a, b, ab\}$. Furthermore, we can now look for other such subgroups by conjugating this one. When we do this we notice that all conjugations by elements not belonging to H give something besides H, i.e. the normalizer $N(H) = H$. [Furthermore, we can easily realize that the same is true for conjugates of H, i.e. $N(gHg^{-1}) = gHg^{-1}$ for any $g \in G$]. Thus there are $[G : N(H)] = \frac{20}{4} = 5$ order 4 subgroups conjugate to H.

We claim that this collection of subgroups $\mathcal{H} = \{H = H_1, H_2, H_3, H_4, H_5\}$ is the entire list of order 4 subgroups of G. Suppose that $W = \{a_0 = 1, a_1, a_2, a_3\}$ is any order 4 subgroup of G. Since this is a copy of 0402 we know that if $1 \leq i, j, k \leq 3$ are all distinct then $a_i a_j = a_k$. Now given a specific H_i we can consider all the conjugate subgroups $a_j H_i a_j^{-1}$ for $j = 0, 1, 2, 3$. These will all belong to $\mathcal{H}$ and the process of doing this for all i and j will partition $\mathcal{H}$ into a collection of disjoint subsets where two subgroups in $\mathcal{H}$ belong to the same subset iff they are conjugate in this way by an element from W.

We claim that each such subset (called a W-orbit) will have either 1, 2 or 4 elements. Let $< H_i >$ denote the W orbit of H_i from $\mathcal{H}$. From what we said above we know that $a_j H_i a_j^{-1} = H_i$ iff $a_j \in H_i$. Thus $| < H_i > | = 1$ iff $H_i = W$ and $H_i \cap W = \{1\}$ iff $| < H_i > | = 4$. Since the order of $W \cap H_i$ must divide 4 we know that this intersection cannot have size 3. Thus the only remaining possibility is when (without loss) $H_i \cap W = \{1, a_1\}$. In this case we have that $< H_i > = \{H_i, a_2 H_i a_2^{-1}, a_3 H_i a_3^{-1}\}$. That this set has size 2 follows once we note that $a_2 H_i a_2^{-1} = a_3 H_i a_3^{-1}$. This is clear because conjugation (which is injective on subgroups) of both $a_2 H_i a_2^{-1}$ and $a_3 H_i a_3^{-1}$ by a_2 gives, (using that $a_2^2 = 1$ and $a_2 a_3 = a_1$) H_i.

Therefore $\mathcal{H}$, which has size 5, is partitioned into disjoint W-orbits of sizes 1, 2, and 4. Therefore, there must be an orbit of size 1. It is the H_i in this orbit which must equal W. Thus there are exactly 5 subgroups of order 4, all are of type 0402, and they are all conjugate to one another.

Plans for the future

Based on the registration information we have received, it now appears that approximately 41 schools throughout the USA and Canada have expressed an interest in using FGB. In addition to this, if future windows environments stop supporting the DOS 16 bit format used by ESG, then FGB will become an increasingly important alternative for those who wish to teach in this manner. It is our sincere hope that FGB can live up to the expectations of those educators who will want to use it in the future. We are trying to understand what new features need to be added as well as what existing bugs need to be fixed. Based on the comments we have received so far, we have constructed the following list of most commonly considered improvements. These are listed from the most routine to the most ambitious:

- The ability to see the formation of subgroups in a step by step fashion but without sacrificing the speed that currently exists for those who don't need this.
- The option to record homomorphism formulas in the notes by right clicking as in the **Group table** and **Subgroup table** tabs.
- More on line documentation for using the program, especially intended for the beginning student.
- The addition of a calculator tab to each group window. This would allow the user to make calculations in the group and solve equations.
- The ability to calculate the orders of elements more directly.
- Inclusion in the library all Abelian groups of size 40 or less and possibly Cayley tables which are not groups.
- A web-based version of FGB. This would eliminate the need to use a windows platform for FGB and thus make the software available to anyone with access to the internet.
- The ability to perform constructions with groups such as direct sums or even semi-direct products.
- A graphical interface that would allow the user to grasp the geometric interpretations of some of the groups (like the dihedral groups), when this is appropriate.
- Improved connections of FGB to the applied areas of group theory, especially cryptography and combinatorics.
- Automated capabilities to search the library for common features of groups or the occurrence of specified phenomena.

For students, one of the excellent features that a program like ESG or FGB offers, is the opportunity to survey the features of a wide variety of groups in an attempt to find their similarities and differences. These similarities can lead to conjectures and subsequent research by the student. These aspects are a big part of the teaching style exemplified in [7] or the results obtained in REU projects such as those of [1,6]. We have been told that it would be helpful if it would be possible to automate this process and look for various features of many groups in the library and keep statistics on what is found. Thus some ability to write and perform macros on one or more groups in the library might be useful. However, it is not at all clear if enough functionality would be possible and since Magma [5] and GAP [3] currently have this capacity, incorporating it may not be worth the effort.

The authors are proud of FGB and we feel strongly that both in its current form and in future releases, it has a great potential to enhance the beginning group theory classroom. We sincerely hope that, after a careful examination of this paper and use of the program, that you will agree. However, for FGB to continue to grow effectively, we depend heavily on user input. Please write to Ed Keppelmann at *keppelma@unr.edu* with your comments, questions, wishes, and concerns. Given the existence of continued demand for the software, which is monitored through the free registrations required by all users (or instructors on behalf of their classes), new releases of FGB should be a not infrequent event. We are committed to making sure that the software always remains free and relevant for those individuals who wish to use it.

REFERENCES

[1] Belcastro, S. and Sherman, G., (1994) "Counting centralizers in finite groups", *Mathematics Magazine* **67**, *no.* 5, 366-374.
[2] Hungerford, T., (1997) *Abstract Algebra, An Introduction*, 2nd Ed., Saunders College Publishing.
[3] Schnert, M., *GAP: Groups, Algorithms, and Programming*, fifth edition, Lehrstuhl D fr Mathematik, Rheinisch Westflische Technische Hochschule, Aachen, Germany, *http://www-groups.dcs.st-and.ac.uk/ gap/gap.html*.
[4] Geissinger, L., (1989) *Exploring Small Groups: A Tool for Learning Abstract Algebra*, Saunders College Publishing.
[5], *MAGMA*, Computational Algebra Group of the School of Mathematics and Statistics, University of Sydney, *http://www.maths.usyd.edu.au:8000/u/magma/*.
[6] Mitchell, C., and Sherman, G., (1995) "Counting products of triples in finite groups", Proceedings of the 26th Southeastern Conference on Combinatorics, Graph Theory and Computing, *Congressus Numerantium* **109**, 129-134.
[7], Parker, Ellen M., (1996) *Laboratory Experiences in Group Theory*, Mathematical Association of America, Providence RI.

Edward Keppelmann
Department of Mathematics AB601 MS084
University of Nevada Reno
Reno, Nevada, 89557
keppelma@unr.edu

Bayard Webb
c/o Edward Keppelmann
Department of Mathematics AB601 MS084
University of Nevada Reno
Reno, Nevada 89557
bayard@bsw.reno.nv.us