Computer Applications in Teaching Abstract Algebra

Muzaffer OKUR
Mathematics Education, Erzincan University,
Erzincan, Turkey
E-mail: mokur@erzincan.edu.tr, Phone #: 446 2240089

Ramazan DİKİCİ
Mathematics Education, Atatürk University
E-mail: rdikici@atauni.edu.tr

Vehbi Aytekin SANALAN
Computer Education and Instructional Technology, Erzincan University,
E-mail: sanalan@erzincan.edu.tr

Enver TATAR
Mathematics Education, Atatürk University
E-mail: entatar@atauni.edu.tr

Abstract

Many mathematical concepts are abstract and learning these concepts requires advanced level cognitive activity. As a result, conceptual learning of this kind is complicated and requires more student effort. It is increasingly possible to demonstrate and explain many of the abstract concepts with proper computer technology. Through the use of computer software many of the abstract concepts become easier for students to grasp. Computers can be powerful aids for the teaching and learning of mathematics when the appropriate software is used. In this study, various software packages (GAP, ISETL, ESG, FGB, etc.) are evaluated based upon their applicability to teaching and learning mathematics, especially abstract algebra. Additionally, this paper considers the use of the computer software Finite Group Behavior (FGB) in teaching an introductory group theory course at the university level. FGB is discussed in detail and some concrete examples are presented for learners and educators.

Keywords: Mathematics education, abstract algebra, computer assisted instruction, software, FGB

1. Introduction

This century is often called the "Age of Information" because advances in communication and computer technology continually change and facilitate our lives. In order to provide excellence in education and adapt to new developments, it is necessary to implement new technological developments in educational programs. One of the most valuable developments of our era is the developments in information and communication technologies. The computer is a multi–functional device offering unique opportunity for teaching and learning. When used in instruction technology creates an atmosphere that supports students' educational experiences. Current developments in computer technology affect all global, cultural, and economic life as well as educational settings (İşman et al., 2004). Furthermore, computers are helpful in differentiating roles of students and teachers, implementing equal standards in instruction. Education software can also equalize and encourage student understanding and meaningful learning for all students within the constructivist approach. Teacher-centered instruction is spontaneously becoming student-centered when a multiple intelligence atmosphere is launched in the educational cycle through the use of computers (Forcier, 1996).

1.1. Computer and Mathematics Education

The computer is being used in the teaching of mathematics with growing interest. However, the user characteristics and many other variables define how the technology used in mathematics teaching. At first, teachers and educational researchers thought of the computer as a supportive material, like an overhead projector, slides or a television. Computer applications were initially used for presenting colorful electronic pages with colorful graphics and simple calculations rather than guiding students to construct their own knowledge. This failed to offer a strong base for meaningful learning and effective teaching practices (Güven & Karataş, 2003). Some studies even report negative outcomes from these traditional uses of computers in teaching. An example of traditional use of computers in teaching algebraic concepts and processes can be using computers as calculators where students can see the result of a simple calculation. According to Norcliffe (1996), students would lose their algebraic calculation abilities when they use computers for simple calculations only.

Mackie (2002) expressed that the students may reach a point where they lack the understanding of what they are doing, when they merely type some necessary keys to process problems in a computer software program. Generally, the understanding of certain abstract mathematical concepts has always been a struggle. Because the structure of mathematics offers something abstract, learning abstract concepts can be difficult for many students. However, this difficulty can be overcome or reduced by using concrete tools provided by software (Baki, 2002). Technological tools can be used in developing mathematical thinking, problem solving and creativity to build student concentration skills. These software programs discourage rote learning, and prevent students working on exhausting routines. In conjunction with algebra teaching software, the effective use of information processing technology in the teaching of school mathematics is one of the topics primarily discussed among researchers. Adopting new technologies in teaching mathematics can influence the development of educational policy, strategy, and structure (Ersoy, 2003).

The high-tech industries of the new century have made abstract thinking and creative abstract abilities prerequisites for global economic competitiveness. Students can only gain the abilities of abstract thinking once they establish the logical foundations of mathematics. The development of information processing tools in mathematics learning can help students gain problem solving and critical thinking skills. In this process the social constructivist approach should be used in teaching to form skills such as exploration, analysis and algorithms of the logical relations behind the concepts (Hacısalihoglu et al., 2003). A vast body of literature exists reporting the effectiveness of computer-assisted instruction. These studies have shown that student achievement is much higher in classrooms where computers are used than in those without, and that using this approach, the students learn new concepts more effectively (Krishnamani & Kimmins, 2001; Rainbolt, 2002; Van & Dempsey, 2002). In addition, some researchers report that computer assisted instruction led to the development of upper level thinking skills and that the students learn conceptualizing rather than memorizing (Renshaw & Taylor, 2000; Ubuz, 2002).Developing from knowledge theory based on the constructivist philosophy, the use of the information processing may lead to a more fertile and functional teaching environment.

In such an environment, the student can analyze complicated research problems, think about possible solutions, form hypotheses, and make generalizations. The student can draft his own mathematical works using the software. The software serves as a tool for the student to explore the information, concepts and facts in a working scenario prepared by the teacher. The student can participate in the activity using all the means to control his own learning. Derive™ and Mathematica™ are two software programs offering such learning facilities. Most software packages available in the market require very little programming skills. For example: Excel is a software program that can be used in place of pen, paper and calculator. The conversion table and graphics are generated by writing the necessary formula to the electronic table (Baki, 2001). Today there are many computer software programs used for teaching and learning of mathematics (Cannon & Playoust, 1997; Krishnamani & Kimmins, 2001; Baki, 2001; Rainbolt, 2002; Çınar & Ardahan, 2003; Bardzell & Shannon, 2004; Gülcü, 2004). The computer algebra systems such as, Derive™, Mathematica™, Magma™ and Maple™ have brought some innovations in advanced calculations, differential equations and other applied fields (Kulich, 2000).

1.2. Using Computer Algebra Systems in Teaching Abstract Algebra

Today the number of studies related to the teaching of mathematics at all levels is increasing considerably. However, research related to the teaching of mathematics at the university level is still rarely reported. Most of the experimental studies are about analysis, linear algebra and discrete mathematics (Hazzan, 1999). There are, however, a number of new research efforts into the teaching and learning of abstract algebra. Abstract algebra is widely used in computer sciences, physics and chemistry. In addition it provides the foundation for the study of advanced mathematics. Questions that researchers have posed have included: “can all the students learn the concepts more meaningfully” and “what is the best method for teaching?”. These questions led algebra educators to do more research on teaching abstract algebra (Krishnamani & Kimmins, 2001).

One of the most significant challenges in teaching and learning math is how to handle abstract concepts. Algebra plays an important role not only in making the meaning loaded by the notations but also in getting the results easily right after applying a mathematical operation and in generalization of logical arguments (Hacısalihoglu et al., 2003). Current studies on teaching and learning abstract algebra can roughly be divided into two groups: the first group considers the methods of teaching abstract algebra while those in the second group study the development of conceptual comprehension and teaching abstract algebra (Hazzan, 1999). Many problems are identified in the teaching and learning of abstract algebra in the literature (Almeida, 1999; Hazzan & Zazkis, 2005). The researchers have also come to the conclusion that students' mistakes do not arise only from the complexity of the subject but also from weaknesses in the students' mathematical foundations.

The concepts such as group theory in an introduction to abstract algebra courses are particularly difficult for the students to learn and to maintain positive attitudes toward (Campbell & Zazkis, 2002). It is possible to present most of these concepts using computer technology and to make them available for students for further learning. By using this approach most abstract concepts can be concretized and they become easier for students to grasp (Baki, 2002). Today, there are many software programs used in the teaching and learning of abstract algebra. One of the programs is Magma™ Computer software (Quinlan, 2007). Magma™ has a program language directly constructed on basic algebra notations and meaning. In addition, Magma™ is suitable for studying algebra individually unlike Maple™ and Mathematica™ (Kulich, 2000). It has been developed by a team of computer-algebra researchers supported by the Australian Research Union at the University of Sydney. This team also developed Cayley™ software to teach the group theory in the related areas of the theory between 1975 and 1985.

The starting point for the design of Magma™ was Cayley™. While Cayley™ covers a limited area of algebra Magma™ covered the general algebra system. Magma™ is one of the computer software packages that can be used especially for algebra, number theory and geometry. Some well-known categories that Magma™ can be used to teach are finite groups, semi groups, rings, fields and geometric structures. In addition, the program has been used for advanced computer algebra courses, introductions to group theory courses and the teaching of vector spaces (Cannon & Playoust, 1997). Two other of the most common programs used in abstract algebra are Matlab™ and Maple™ (Quinlan, 2007). Maple™ computer algebra system was originally designed at Waterloo University in Canada. Charlwood (2002), studied teaching symmetric and matrix groups in an introductory abstract algebra course and indicated that the teaching approach enriched by Maple™ computer working pages contributed to the development of proofing abilities in students.

Mackiw (2002) stated that Matlab™ is useful for defining the group relations, calculating the order of elements and showing the elements of the sub-groups similar to Maple™. He also expressed that by using these programs students have a chance to examine the non-commutative group types which they rarely meet. Both Matlab™ and Maple™ can be used in groups, rings and finite fields in teaching abstract algebra and many other related areas such as cryptograph, coding theory and Polya counting theory Klima et al., 2000; 2007). Other computer software such as ESG, FGB, GAP, ISETL have been developed only for teaching of abstract algebra. Another software used for abstract algebra is GAP (Groups, Algorithms and Programming), a useful calculation tool for structuring hardware and most operating systems (Gallian, 2010). GAP is used mostly by mathematics researchers. Rainbolt (2002) reported that students learn algebraic concepts more quickly when this tool is used. He also stated that it allows students to test what they learned and explore some algebraic concepts in groups. Computer activities can be divided into two categories.

The first category involves activities to guide students to translate patterns into equations. In most cases students can achieve well-known algebraic principles or rules by doing these types of activities. Algebraic structure can be discovered by students even when they lack the proper definitions, and this is just one learning outcome of this practice. Another category of computer applications promotes deeper understanding of a concept by providing numerous examples. GAP enables students to investigate groups and other complex algebraic structures. Furthermore, GAP seems to arouse students' interest and curiosity and can be used as a mean to motivate and engage them in class (Rainbolt, 2002). ISETL (for DOS) or ISETLW (for Windows) is the other type of computer software that is commonly available, and its Java version is up due in 2010 (Kirschmeyer, 2010). This software gives students the opportunity to set the logical conditions on elements, form a group and then explore algebraic properties such as order, cosets, quotients, homomorphism and isomorphism. It provides students with an active learning atmosphere similar to GAP.

In their book titled "Learning Abstract Algebra with ISETL", Dubinsky and Leron (1994) explain that the knowledge is based on constructivist belief that the students need to be engaged in mental activities, and this vision is shared by other researchers in more recent studies (Selden, 2005). These computer-supported activities establish an experiential base for any future verbal explanations before students can make sense of any presentation of abstract mathematics. Additionally they have the opportunity to compare these explanations with their cognitive experiences. This approach is based on extensive theoretical and empirical research as well as on the substantial experience of the authors in teaching abstract algebra. The main activities in an algebra course is computer constructions, specifically, small programs written in the programming language ISETL. The main tool for reflection is work in teams of 2-4 students where the activities are discussed and debated. Because of the similarity of ISETL expressions to standard written mathematics, there is a little initial information about using ISETL. ACE (Activities, Class discussion material, Exercises) is a teaching circle created by the researchers using ISETL-based teaching methods.

The learning cycle divides the course into weekly blocks. Each week students meet in a computer lab on certain days and in a regular classroom for the rest of the week. Accompanying textbook is written in an informal, discursive style, closely relating definitions and proofs to the constructions in the activities. Many mathematics educators including the first author have used the ISETL in abstract algebra teaching. There is an agreement among researchers that ISETL can be an effective learning tool (Pesonen & Malvela, 2000; Khrishnamani & Kimmins, 2001; Smith, 2002; Okur, 2006; Weller et al., 2002). The FGB (Finite Group Behavior), an upgraded Windows™ version of ESG (Exploring Small Groups) was originally written by Edward Keppelmann. This software has some similar features with the ESG which runs on DOS (Kulich, 2000). It is used for teaching the basic features of the group theory; the finding of group, subgroups, normal subgroups, factor groups and order of elements. The FGB checks whether an algebraic structure with its elements supplies the group features with the help of operation tables. Students can see the subgroup and factor groups of a group on their screen with color coding of the group tables (Perry, 2004).

Keppelmann and Webb (2002) pointed out that the FGB can be used with novice learners to drill and practice with concrete examples of groups. The program runs on a library of all finite groups up to isomorphism of order 16 or less and all non-abelian groups of order 40 or less. The table for the groups can be saved as text files with a specific structure and users can keep notes on the features they discover. Each discovery is then stored in a separate place in the file along with the group information. Many groups can be studied at the same time for comparison (FGB, 2010). The FGB program has seven tabs for the analysis of each group:

1) Axiom check: It allows verification of the axioms for any given group table: associativity, existence of an identity, and inverses.

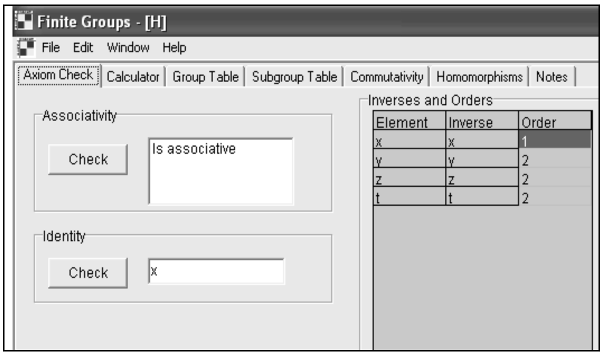

Figure 1. Checking the group axioms for group H

2) Calculator: This is used to calculate powers and inverses for elements as well as complicated calculations involving multiple elements.

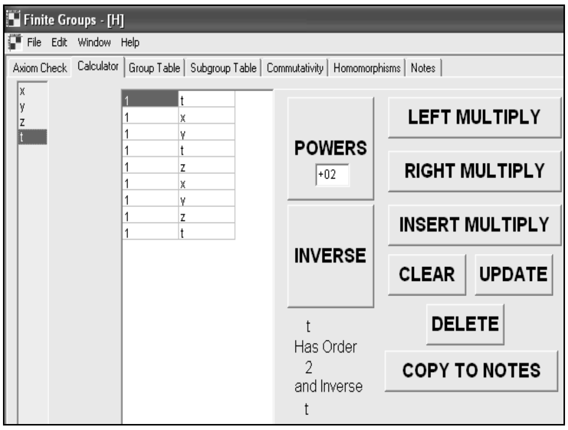

Figure 2. Calculation of powers and inverses of elements in group H

3) Group Table: This tab shows the Cayley™ table for the group. It also enables reordering or renaming of the elements.

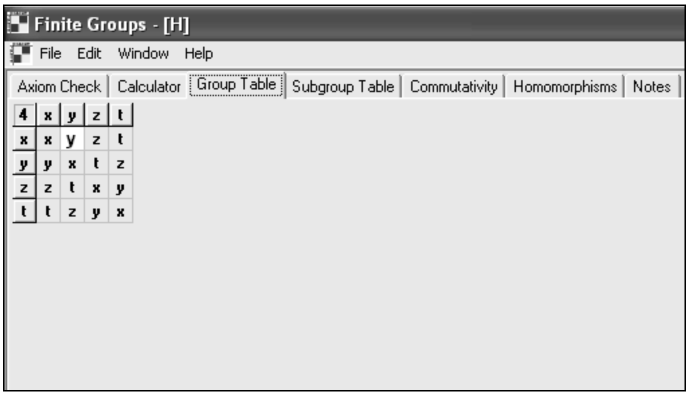

Figure 3. Group table for H

4) Subgroup table: This tab is to form the subgroups of an open group. Cosets for groups can be viewed in a special colored framework which allows one to easily see whether the resulting collection of cosets forms a group or not. Subgroups can be conjugated by arbitrary elements and both quotient groups and subgroups can be exported to create their own group files. Notes can be made to record information about the subgroup and used for later reference.

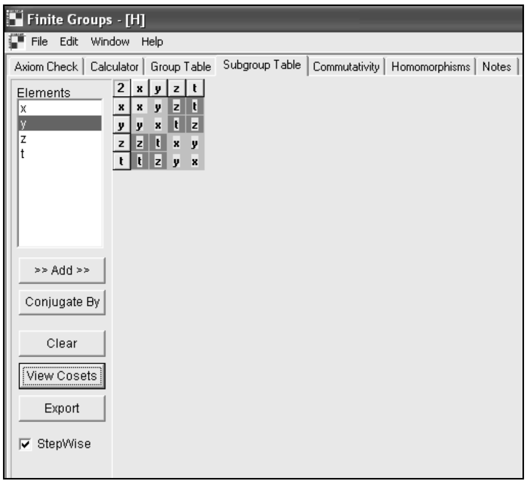

Figure 4a. Calculating the cosets of group H

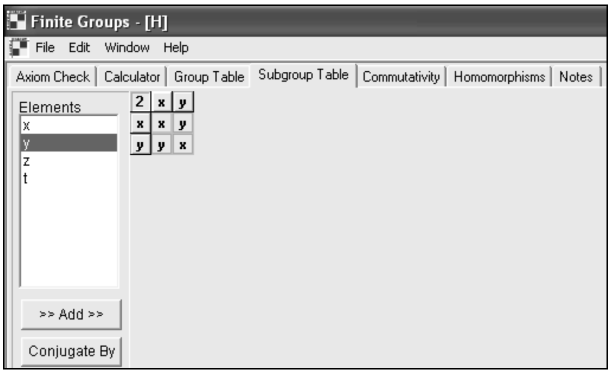

Figure 4b. Finding a subgroup of group H

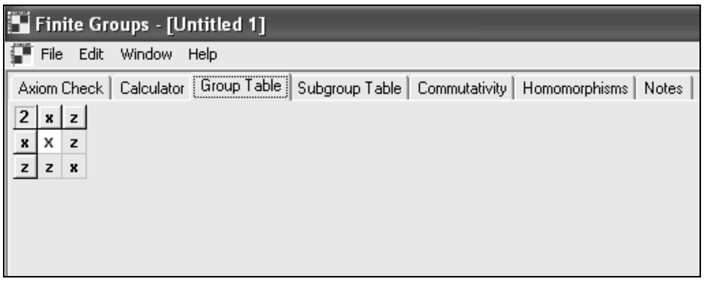

Figure 4c. Determining a quotient group of group H

Figure 4. Calculating cosets, subgroups and quotient groups of group H

5) Commutativity: Using this tab, the center of the group and centralizers of elements can easily be calculated.

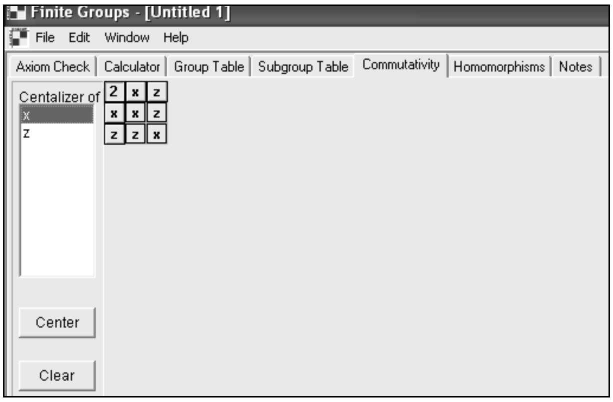

Figure 5. Calculating the center and centralizers of group H

6) Homomorphisms: This tab allows one to construct homomorphisms between any two group files.

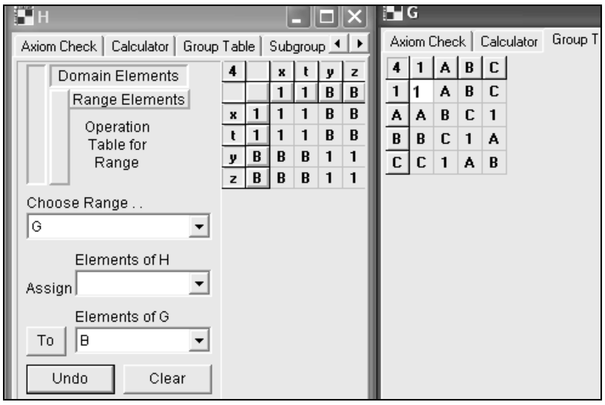

Figure 6. Constructing a homomorphism between groups H and G

7) Notes**:** Features of a group can be recorded here. The notes also contain explanatory information about the groups when this is appropriate including an explanation of the notation for the descriptive elements.

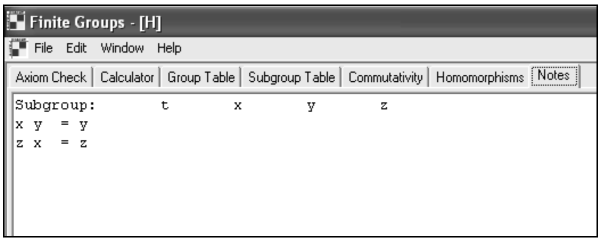

Figure 7 Storing exploratory and descriptive information about group H

2. Conclusions

Mathematics, especially the concepts of abstract algebra are often considered to be difficult to understand, since the concepts are abstract. They need to be concretized by examples in order to be conceptualized more easily. Most abstract concepts taught in courses of abstract algebra can be concretized by means of mathematical software.

The studies indicate that using computer assisted instruction methods can clearly make a great contribution to students' understanding of conceptual relations, argumentation and mathematical proofs. These computer programs provide many benefits in learning abstract algebra concepts. For instance, some software programs such as Matlab™, GAP, ISETL and FGB are useful for visually defining the group relations, calculating the order of the elements and showing the elements of sub-groups. The students using these software packages have also a chance to examine non-commutative group types . Students can also can set the logical conditions on elements to form a group and then explore algebraic properties i.e. order, cosets, quotients, homomorphism and isomorphism. Using computer programs can create active learning environments and opportunities for testing what students have learned.

Classes can be structured to explore abstract algebra concepts as individual students or in groups. As we understand from the literature, computer applications can be critical to achieving broader goals such as improving problem solving and higher order thinking skills, as well as developing self-regulation and collaborative learning abilities. This may eventually improve student achievement by increasing comprehension rather than rote memorization. It is important to maintain student attention, motivation and curiosity. Using computer applications to teach abstract algebra can promote meaningful learning of the concepts that can be difficult to grasp otherwise. When the subject is abstract it is more difficult to establish meaningful connections with what is already known. Abstract algebra software combined with proper teaching techniques can offer prosperous learning environments where students engage in discovery, control their learning, and work collaboratively in groups.

References

Almeida, D. F. (1999). Visual aspects of understanding group theory, International Journal of Mathematical Education in Science and Technology, 30(2), 159-166.

Baki, A. (2001). Bilişim teknolojisi ışıgı altında matematik egitiminin degerlendirilmesi, Milli Egitim Dergisi, 149, 26-31.

Baki, A. (2002). Ögrenen ve Ögretenler İçin Bilgisayar Destekli Matematik, İstanbul, Turkey: Ceren Yayın Dagıtım.

Bardzell, M.J., & Shannon, K.M. (2004). The PascGalois Project: Visualizing Abstract Algebra/Mathematics, Joint Meetings of the American Mathematical Society and the Mathematical Association of America.

Campbell, S., R., & Zazkis R. (2002). Learning and teaching number theory: research in cognition and instruction, Ablex Publishing.

Cannon, J., & Playoust, C. (1997). Using the MAGMA computer algebra system in abstract algebra courses, Journal Symbolic Computation, 23, 459-484.

Charlwood, K. (2002). Some uses of Maple in the teaching modern algebra, Innovations in Teaching Abstract Algebra, Mathematical Association of America, 91-96.

Çınar, C., & Ardahan, H. (2003). Excel ile matematik (Bilgisayar Destekli Matematik Ögretimi), Konya, Turkey: Dizgi Ofset Matbaacılık.

Dubinsky, E., & Leron, U. (1994). Learning abstract algebra with ISETL, Springer-Verlag.

Ersoy, Y. (2003). Teknoloji destekli matematik egitimi-1: gelişmeler, politikalar ve stratejiler, İlköğretim-Online Dergisi, 2(1), 18-27.

FGB 3.0, (2010). An Overview Of Features, available: http://wolfweb.unr.edu/homepage /keppelma/fgb3over.pdf (Retrieved 17 May 2010).

Forcier, C., R. (1996). The computer as a productivity tool in education. Prentice-Hall, Inc. A Simon & Schuster Company in United States of America.

Gallian, J., A. (2010). Contemporary Abstract Algebra, Seventh Edition, Brooks/Cole, Cengage Learning Press.

Gülcü, A. (2004). Mathematica 5 bilgisayar destekli matematik, Ankara,Turkey: Nobel Yayın Dagıtım.

Güven, B., & Karataş, İ. (2003). Öğrenen ve öğretenler için bilgisayar destekli matematik, İlköğretim-Online Dergisi, 2(1), 52–53.

Hacısalihoglu, H. H., Mirasyedioglu, Ş., & Akpınar, A. (2003). Matematik ögretimi, Ankara, Turkey: Asil Yayın Dagıtım.

Hazzan, O. (1999). Reducing abstraction level when learning abstract algebra concepts, Educational Studies in Mathematics, 40, 71–90.

Hazzan, O., & Zazkis, R. (2005). Reducing abstraction: The case of school mathematics, Educational Studies in Mathematics, 58, 101-119.

İşman, A., Çaglar, M., Dabaj, F., Altınay Z., & Altınay F. (2004). Attitudes of Students toward Computers, The Turkish Online Journal of Educational Technology,3(1), Article 2. [On-line].

Keppelman, E., & Webb, B. (2002). Learning beginning group theory with Finite Group Behavior, Innovations in Teaching Abstract Algebra, Mathematical Association of America, 45-54.

Kirschmeyer, J., (2010). ISETLJ Home Page, available: http://raider.muc.edu/~kirchmjf/isetlj/isetlj.html. (Retrieved 17 May 2010).

Klima, R. E., Sigmon, N., & Stitzinger, E. (2000). Applications of abstract algebra with Maple, CRC press LLC.

Klima, R. E., Sigmon, N., & Stitzinger, E. (2007). Applications of abstract algebra with Maple and Matlab, Taylor&francis Group LLC.

Krishnamani, V., & Kimmins, D. (2001). Using Technology as A Tool in Abstract Algebra and Calculus: The MTSU Experience, http://archives. math.utk.edu/ICTCM.

Kulich, L.T. (2000). Ideas for teaching and learning: Computer algebra in abstract algebra, The International Journal of Computer Algebra in Mathematics Education, 7(3), 213–219.

Mackie, D. (2002). Using computer algebra to encourage a deep learning approach to calculus, 2nd International Conference on the Teaching of Mathematics, at the Undergraduate Level, July 1-6, Hersonissos, Crete , Greece.

Mackiw, G. (2002). Experiments with finite linear groups using MATLAB, Innovations in Teaching Abstract Algebra, Mathematical Association of America, 85–90.

Norcliffe, A. (1996). The revolution in mathematics due to computing, TALUM newsletter (4), Mathematical Association, www.bham.ac.uk/ctimath/talum /newsletter/talum4.htm.

Okur, M. (2006). Teaching isomorphism concept through algebra software packages, PhD diss., Atatürk University, Erzurum-Turkey.

Perry, A. B. (2004). A discovery oriented technology enhanced abstract algebra course, Education, 124(4), 694-698.

Pesonen, M., & Malvela, T. (2000). A reform in undergraduate mathematics curriculum: more emphasis on social and pedagogical skills, International Journal of Mathematical Education in Science and Technology, 31(1), 113-124.

Rainbolt, J. G. (2002). Using GAP in an abstract algebra class, Innovations in Teaching Abstract Algebra, Mathematical Association of America, 77–83.

Renshaw, C. E., & Taylor, H. A. (2000). The educational effectiveness of computer-based instruction, Computers and Geosciences, 26(6), 667-682.

Selden, A., (2005). New developments and trends in tertiary mathematics education: or, more of the same?, International Journal of Mathematical Education in Science and Technology, 36, 131-147.

Smith, R. S. (2002). Using ISETL and cooperative learning to teach abstract algebra: an instructor's view, Innovations in Teaching Abstract Algebra, Mathematical Association of America, 71-76.

Quinlan, J. E. (2007). Profiles of software utilization by university mathematics faculty, PhD diss., The Ohio State University.

Ubuz, B. (2002). Development of calculus concepts through a computer based learning environment, Proceedings of the 2th International Conference on Teaching of Mathematics, 1, 1-10.

Van, E., & Dempsey, J. (2002). The effect of competition and contextualized advisement on the transfer of mathematics skills in a computer-based instructional simulation game, Educational Technology Research and Development, 50(3), 23-41.

Weller, K., Montgomery, A., Clark, J., Cottrill, J., Trigueros, M., Arnon, I., & Dubinsky, E., (2002). Learning Linear Algebra with ISETL, available: http://homepages.ohiodominican. edu/~cottrilj/datastore/linear-alg/LLAWI-P3.pdf. (Retrieved 18 May 2010).