

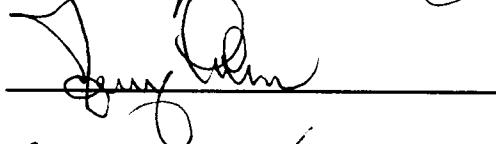
To the Graduate Council:

I am submitting herewith a thesis written by John Mitchell Barron entitled "Rhythmic Recognition and Dictation: A Microcomputer-Based Approach." I have examined the final copy of this thesis for form and content and recommend that it be accepted in partial fulfillment of the requirements for the degree of Master of Music.



Donald Pederson, Major Professor

We have read this thesis and
recommend its acceptance:



Accepted for the Council:



Vice Provost
and Dean of The Graduate School

STATEMENT OF PERMISSION TO USE

In presenting this thesis in partial fulfillment of the requirements for a Master's degree at The University of Tennessee, Knoxville, I agree that the Library shall make it available to borrowers under rules of the Library. Brief quotations from this thesis are allowable without special permission, provided that accurate acknowledgement of the source is made.

Permission for extensive quotation from or reproduction of this thesis may be granted by my major professor or, in his absence, by the Head of Interlibrary Services when, in the opinion of either, the proposed use of the material is for scholarly purposes. Any copying or use of the material in this thesis for financial gain shall not be allowed without my written permission.

Signature John Mitchell Banon

Date 2/19/88

**RHYTHMIC RECOGNITION AND DICTATION:
A MICROCOMPUTER-BASED APPROACH**

**A Thesis
Presented for the
Master of Music
Degree
The University of Tennessee, Knoxville**

**John Mitchell Barron
March 1988**

ACKNOWLEDGMENTS

During the course of this study, Professors Donald Pederson and Keith McClelland provided invaluable support and assistance. I would like to express my sincere gratitude for their efforts.

I also offer my humble thanks to my wife, Belinda, for her unflinching support during the completion of this project and many others throughout the course of my educational career.

ABSTRACT

During the last twenty years computer-assisted instruction (CAI) in music education has progressed from the experimental, in which programs were developed and presented on mainframe computers, to integrated settings in which microcomputers are used on a more cost-effective basis to develop and deliver courseware. In response to tremendous growth in this area, commercial programmers have become significant contributors in courseware development.

A recent survey of CAI products in music education by Moorhead (1987) reveals that out of eighty-nine programs only five are concerned exclusively with rhythmic training. While these programs are well designed products, a need exists for programs containing a variety of specialized features for education. This project was undertaken in order to address that need.

During the completion of this study, twenty-three programs were developed, producing a series of ten lessons in rhythmic recognition and dictation. All programs are designed to operate on any computer in the "Apple II" line and were coded in Applesoft BASIC. The programs make extensive use of The Musician's Toolbox by Daley, Kolosnick, and Turner (1987) for sound reproduction and graphics creation.

The rhythmic exercises developed in this study utilize a pool of 375 short melodic examples and provide immediate feedback in the form of evaluation and diagnosis of the user's response. Tests are competency based and are structured in a combined drill-test format.

TABLE OF CONTENTS

CHAPTER	PAGE
INTRODUCTION.	1
I. A REVIEW OF THE RELATED LITERATURE.	3
A Brief History of Materials Available for Computer-Assisted Instruction in Music Education	3
A Review of Three Representative Programs	6
The Rhythmic Dictator	7
Hardware Requirements	7
Learning Strategies and Materials	7
User-Program Interaction.	8
Record Keeping.	9
Evaluation.	9
The Music Class: Rhythm.	9
Hardware Requirements	9
Learning Strategies and Materials	10
User-Program Interaction.	11
Record Keeping.	12
Evaluation.	12
Music Theory Vol. 1: Rhythm.	12
Hardware Requirements	12
Learning Strategies and Materials	13
User-Program Interaction.	14
Record Keeping.	15
Evaluation.	15
II. PROGRAM DEVELOPMENT	17
Teaching Strategies	17
Program Limitations	18
The Musician's Toolbox.	20
The Shell Concept	21
Record Keeping and Security	21
III. DATA DESIGN	23
The Basic Rhythmic Pattern Set.	23
Pitch Content of Melodic Examples	25
Rhythmic Content of Melodic Examples.	26
IV. PROGRAM DESIGN.	29
Control of Program Flow	29
Screen Design	30
The Problem Presentation Sequence	32
Error Messages.	32
The Recognition Lesson Problem Algorithm.	32

CHAPTER	PAGE
IV. (Continued)	
The Dictation Lesson Problem Algorithm.	34
The Dictation Lesson Playback Algorithm.	36
The Alternate Character Set	38
Utility Programs.	40
Data Creation Programs.	41
Data Validation Programs.	41
V. SUGGESTIONS FOR IMPLEMENTATION.	42
Record Keeping Additions.	42
Data Revision	42
Algorithmic Changes	43
BIBLIOGRAPHY.	44
APPENDIXES.	47
APPENDIX A. THE ALTERNATE CHARACTER SET.	48
APPENDIX B. MELODY BANK 1.	50
APPENDIX C. MELODY BANK 2.	54
APPENDIX D. MELODY BANK 3.	58
APPENDIX E. MELODY BANK 4.	62
APPENDIX F. MELODY BANK 5.	66
VITA.	70

LIST OF FIGURES

FIGURE	PAGE
1. The Basic Rhythm Pattern Set.	24
2. Typical Harmonic Configurations	25
3. The Primary Method of Pattern Combination	27
4. The Secondary Method of Pattern Combination	28
5. Syncopated Combinations	28
6. The Truncation and Extension Functions of the Playback Algorithm	37
7. The Relative Proportion of Beamed Rhythmic Patterns	39

INTRODUCTION

The use of computer-assisted instruction (CAI) in music education has grown tremendously in the last ten years. In response to this growth, the focus of software development has shifted from the educational to the commercial field as the market place provides an economic incentive to individuals.

A recent listing of music education software by Moorhead (1987) lists some 89 commercial programs currently available. While it does not represent an exhaustive compilation of available materials, it represents the majority of significant music education programs.

A survey of this listing reveals that the overwhelming majority, over 94 percent, focus entirely on music theory or aural training in melodic or harmonic exercises. Programs concerning rhythmic training are included, but their number comprises less than six percent of the total.

Of the five discrete rhythmic training programs listed, three form a series and, in actuality, represent a single unit with three difficulty levels, leaving only three rhythmic training programs of significance to meet the needs of education. Three other rhythmic training lessons are included in the listing, but are not considered here because they form part of larger comprehensive series and are consequently limited in scope.

It is apparent that the development of courseware offering rhythmic training has not received the attention afforded other areas. In response to this lack of emphasis, this study was undertaken.

The programs listed in the survey and programs later reviewed in this paper are well-designed products tailored for a specific educational level. They offer a variety of useful features, including but not limited to the following:

1. Competency-based testing of users.
2. Provisions for automatic record keeping for a large number of students.
3. Data and presentation methods suitable for the intended user population.
4. Efficient teaching strategies incorporating memory, notational and aural elements.
5. Evaluation and diagnosis of the user's response.

While these features may be found individually in a number of programs, this project was designed to provide these and other features in a flexible, cost-effective rhythmic training program.

CHAPTER I

A REVIEW OF THE RELATED LITERATURE

1. A Brief History of Materials Available for Computer-Assisted Instruction in Music Education

The first era in the development of computer-assisted instruction (CAI) in music education began in the late 1960s with the first experimental studies of ear training materials. In 1967 Kuhn and Alvin (Eddins, 1981) conducted a study at Stanford University to develop a system to judge the pitch accuracy of vocal melodies. An IBM 1620 computer was used to control a pitch extraction device and to detect inaccuracies in melodic patterns sung into a microphone. Initial results were encouraging, and a number of areas for future research were identified, including subject-equipment interaction and curriculum planning.

In 1969 at Pennsylvania State University, Deihl (Eddins, 1981) completed a project creating computer-assisted ear training methods for instrumentalists. An IBM 1500 computer was used to coordinate the display of musical examples with pre-recorded tapes. Students were trained to recognize discrepancies in the displayed and aural versions of examples presented. In a second study, Diehl and Zeigler (Eddins, 1981) expanded the project and developed criterion-referenced tests in listening and performance. Pre-test and post-test scores revealed significant gains in both performance skills and aural discrimination.

Investigation of CAI in rhythmic exercises began in 1972 at the University of Illinois. Placek (1974) implemented a lesson designed for students in elementary education covering knowledge of rhythmic notation and the relation of notation to aural rhythmic patterns. Designed and tested on the PLATO system, this project had certain technical problems concerning audio reproduction, but the results of this experimental study confirmed the validity of CAI in music education.

Delivery of CAI materials in music education began on a regular basis in 1973 at the University of Illinois, with the availability of twenty-five programs using the PLATO system that included a series of instrumental lessons developed by Peters (Eddins, 1981). By 1976 a series of music theory lessons had been developed and implemented as a course requirement.

During the period from 1974 to 1979 the GUIDO system of ear training drills and practice routines was created and implemented by Hofstetter (Arenson and Hofstetter, 1983) at the University of Delaware. Developed on a Burroughs 6700 computer and Tektronix Graphics terminals, it was later transferred to the PLATO system (Arenson and Hofstetter, 1983); the GUIDO system was originally conceived as a research device for the investigation of student learning and perceptual patterns but soon proved to be effective in teaching aural skills.

All of the previously cited studies were conducted using large mainframe computers. During the latter half of the 1970s, advances in electronic technology made available an economic alternative: the

microcomputer. As an example, in 1979 Hofstetter (1979) stated that relative inexpense of a GUIDO system with eight PLATO terminals was only \$30,800 per annum for operation and maintenance.

Microcomputers such as the "Apple II," "Atari," "Commodore PET," and "TRS-80" became commercially available, and soon music education software was developed. Perhaps as a reaction to the small amount of memory available (typically 8K to 64K) or the lack of suitable sound generation hardware, stand-alone devices presenting aural skills exercises such as the TAP Master (produced by Editions Ad Lib) were marketed. While these devices incorporated microprocessors, they were not actually computers and lacked programming flexibility. Consequently, they did not achieve widespread use.

A clear trend from the beginning of the 1980s to the present is the use of microcomputers for the delivery of CAI materials. In the first part of this decade, most software in use was developed on mainframe computers but in recent years the tendency has reversed.

Technological advances such as MIDI (musical instrument digital interface) and software development aids such as The Musician's Toolbox (Daley, Kolosnich, and Turner, 1987) are representative of changes in the courseware development environment that have allowed both educational institutions and commercial programmers to produce CAI programs in abundance. A recent listing of products available, Coda: The New Music Software Catalog (Kantor, 1987), contains over 600 items of which 108 are education programs. All products listed are designed to operate on microcomputers, and most were developed on microcomputers.

A recent survey of postsecondary schools conducted at the University of Arizona by Rumery (1986) in 1985 showed that from over 100 respondents, a significant majority have music computer facilities or plan to have them in the near future. The results also indicate that this trend is expected to continue.

The use of CAI materials in music education has grown tremendously in the last twenty years, and evidence shows that its use will continue to expand and become an increasingly important part of the instructional delivery system at all levels of music education. As Hofstetter (1979) states:

By properly harnessing this technology, music educators can make great progress toward improved instructional strategies and delivery techniques. Music educators can collectively develop effective curriculums and distribute them across the general population, perhaps resulting in a keener public awareness and understanding of music and music education.

2. A Review of Three Representative Programs

The three programs selected for review in this study are typical of products commercially available and currently in use. In order to show the similarities and differences in materials and methods of presentation, programs designed for the primary, secondary, and tertiary educational levels are represented.

A standardized format covering essential areas was devised. Areas covered include:

1. Hardware Requirements;
2. Learning Strategies and Materials;
3. User-Program Interaction;

4. Record Keeping;
5. Evaluation.

The Rhythmic Dictator

Hardware Requirements

The Rhythmic Dictator (Parr, Shrader, and Williams, 1982) is designed to run on an "Apple II Plus" or "Apple IIe" computer with at least a 48K memory capacity and a Disk II drive. The Micro Music Inc. DAC music card is required for sound generation; playback capabilities include the Apple speaker and headphones.

Learning Strategies and Materials

As indicated in the user guide, the Rhythmic Dictator is designed for a target population extending from fourth grade to adult.

All examples presented are in duple meter and are monotonic. Examples are computer generated, by using a weighted routine to form random combinations from a set of ten patterns comprised of one, two, three, and four attacks. Variation in the pattern set is achieved by metrically displacing the initial attack. Quarter, eighth, and sixteenth note durations are used but are never employed in combination.

Examples are from two to four patterns long and are employed with three tempos containing minor random fluctuations. (Tempos cover an approximate range of 80 to 108 beats per minutes.) These two parameters, plus an increasing stress on syncopated patterns, are utilized to automatically increase the program difficulty level if the

user's score is above 70 percent and to decrease difficulty if the user's score is below 70 percent. This feature cannot be disengaged.

Each example is played twice initially, and a two beat preparatory measure is included.

User-Program Interaction

At the beginning of the program, the user is asked to select the initial difficulty by choosing both phrase length and pattern difficulty by selective key-presses. The program then displays a single line staff below which the selected series of rhythmic patterns is displayed. The first example is played automatically, and the user then presses the arrow keys to move a pointer in order to locate the first pattern of the phrase. A key-press on the space bar evaluates the selected pattern and, if the selection was correct, displays it on the single line staff. If it is incorrect, a computerized groan is sounded and a prompt is displayed. This process is then continued to the end of the phrase.

To end the work session, the user presses the escape key; similarly, pressing the "Q" key will allow the user to quit working on a particular example.

Scoring consists of the percentage of correct patterns from the total number notated; points are deducted for each listening after the initial two. There is no set number of examples per session; they can continue indefinitely if desired.

Record Keeping

The Rhythmic Dictator does not have provisions for storage of student records. However, the following information is displayed after completion of a lesson:

1. The date of the work session.
2. The starting and ending pattern difficulty.
3. The starting and ending length of examples.
4. The total number and number of correct patterns notated.
5. The user's score.

Evaluation

The Rhythmic Dictator was selected as a program designed for the tertiary educational level. The difficulty level afforded by the program meets this criterion. However, there are certain shortcomings that detract from its suitability:

1. Lack of record keeping provisions.
2. This program is mainly a drill exercise; although it tests users, it does not provide criterion-references tests.
3. Lack of instructor control over material presented; examples are computer generated.

The Music Class: Rhythm

Hardware Requirements

The Music Class: Rhythm (Minnesota Educational Computing Consortium, 1983) is designed to run on any computer in the "Apple II" line with at least 64K of memory and one or more disk drives. The ALS

synthesizer card is an option for enhanced sound quality, but it is not essential.

Learning Strategies and Materials

This series of four programs presents instructional material covering rhythmic notation and exercises in the performance of notated rhythms. Duration values whole through sixteenth notes, including dotted halves and quarters, are employed; all rhythmic examples are monotonic. Four user-selected tempos are employed: 60, 80, 100, and 120 beats per minute. Tempo is not considered as a difficulty criterion and is not included in the evaluation. Detailed information concerning data design was unavailable.

Each of the four lessons presents a test consisting of a seven question written component covering material previously presented and a six example performance component.

The performance section of each test consists of a game-like quiz entitled "Challenge Mr. Metro Gnome." The user is asked to perform rhythms displayed on the work screen by tapping the space bar. A metronomic pulse is not employed.

The written portion of each test is comprised of multiple-choice questions with three possible answers. The user must select the correct answer before moving to the next question, but only the first response is evaluated and graded. A time limit is not employed.

Scoring is calculated as the percentage of correct responses from a total of thirteen; reference criteria are as follows:

1. Below 76 percent -- "Needs Work"
2. 76 to 84 percent -- "Good"
3. Above 84 percent -- "Excellent."

These criteria are preset to these levels; a management option allows the instructor to adjust them as desired.

User-Program Interaction

The series employs a menu-based method of lesson selection, and a screen of instructions explains all key-presses. Prompts detailing the current key-press are included on the work screen. Evaluations of the user's response occurs automatically after entry.

Review and drill options are available during each lesson and are accessed through indicated key-presses. The review option provides a summary of all conceptual material previously covered, and the drill option presents a flexible exercise in dictation.

The drill begins with the user selecting from one to eight rhythm patterns from a displayed menu by using the indicated key-presses. The user can then perform the rhythm by tapping or by having an animated figure ("Mr. Metro Gnome") play the example.

In both the rhythmic drill and quiz, the user has an option to graphically depict the correct or response rhythm on a bar graph by pressing the "D" key. The vertical axis indicates note duration and the horizontal axis indicates the metric placement of the initial attack. The tolerance level of the dictation section of the program can be adjusted by a management option to increase or decrease program difficulty.

Record Keeping

Provisions for recording the scores of 125 students are included as a management option. This feature can be disabled if desired. At the end of an individual lesson the following information is displayed:

1. The number of correct responses on the written portion of the test.
2. The number of correct responses on the performance portion of the test.
3. The total number of correct responses.
4. The user's score and the criteria prompt.

Evaluation

This series is designed for elementary or possibly high school use. The material it utilizes is acceptable for these age groups, but certain graphic elements could possibly be perceived as childish by high school students.

The main technical problem is poor sound quality caused by internal reproduction. Use of the ALF synthesizer card is recommended as a solution.

Music Theory Vol. 1: Rhythm

Hardware Requirements

Music Theory Vol. 1 (Minnesota Educational Computing Consortium, 1983) is designed to operate on any computer in the "Apple II" line with at least 64K of memory. No external sound supports are required.

Learning Strategies and Materials

Music Theory Vol. 1 is a group of eighteen lessons, four of which are concerned with rhythm:

1. "Note Type" -- presents exercises in identifying note names and duration values. It includes whole through sixteenth notes with dotted halves, quarters, and eighths.
2. "Counting" -- presents one measure examples with one note omitted; the user supplies the missing note. Meters used are 2/4, 3/4, 4/4, 5/4, 6/4, 3/8, 6/8, 9/8, 12/8.
3. "Rhythm" -- presents exercises in rhythmic recognition. All examples are in simple quadruple meter, one measure long.
4. "Rhythm Play" -- presents exercises in rhythmic dictation.

All examples are in simple quadruple meter, one measure long.

In each lesson, the user selects the number of problems presented.

The data bank utilized by "Rhythm" and "Rhythm Play" is comprised of sixty rhythmic examples divided into three user-selected levels of difficulty. The first level consists of half, dotted quarter, quarter, and eighth note durations with three, four, or five attacks per measure. The second level consists of dotted quarter, quarter, and eighth note durations with five or six attacks per measure, including some syncopation. The third level consists of quarter, dotted eighth, eighth, and sixteenth note durations with six or seven attacks per measure, including some syncopation.

Examples are selected at random, and once an example has been correctly identified, it is not presented again until all examples have been identified correctly. In the lesson "Rhythm," two distractors

containing the same number of attacks are randomly selected, and pattern playback is monotonic.

User-Program Interaction

A series of help screens containing instructions is provided for each lesson; on-screen prompts detail the key-presses necessary to complete the lesson. Access to each lesson is provided by a central menu program.

Scoring occurs automatically at the end of each lesson and is calculated as the percentage of correct responses from the total number. As stated earlier, in each lesson the user selects the number of problems attempted.

In the "Note Type" and "Counting" lessons, single or double letter abbreviations (W, H, Q, DQ, etc.) and numbers (1, 4, 1.5, etc.) are used to indicate a response. To grade a problem, the user presses the return key. Both lessons evaluate user responses and display the correct response after grading if incorrect. Responses may be altered before grading. After grading, pressing the return key moves the program to the next problem.

The "Rhythm" lesson displays a one-measure pattern, including a four quarter note preparatory measure. The user presses "1," "2," or "3" to hear aural representations that possibly match the displayed pattern. Unlimited listenings of each choice are allowed. After pressing "A," the user again presses "1," "2," or "3" to select the correct aural representation. Pressing the return key grades the problem. If the response is incorrect, the correct choice is

indicated, and the user may play the correct response by pressing "R." If the response is correct, it is so indicated; pressing the return key at this point moves the program to the next problem.

The "Rhythm Play" lesson displays a one measure pattern for dictation, and includes a beat preparatory measure. The user performs the pattern by tapping the "N" key. The preparatory measure is always included in the user's response. Accuracy parameters were not available, but observation indicates that tempo is not a factor in grading; only the relative durations of the user's response are evaluated. Grading occurs automatically after response, and the user's response is displayed after grading. Pressing the "R" key will play the correct pattern; the incorrect response cannot be played. As before, pressing the return key moves the program to the next problem.

Record Keeping

Automatic storage of class records is not available for this program. Tally sheets for each lesson are provided. At the end of each session, the following information is displayed:

1. The number of correct responses.
2. The total number of responses.
3. The percentage of correct responses.

Evaluation

This program is designed for high school or first year college use and is suitable for those levels. Weaknesses include:

1. No record keeping provisions.
2. The small number of examples employed.

3. Only simple quadruple meter is presented in the recognition and dictation exercises.
4. The "Rhythm Play" lesson will not evaluate the user response after the last correct attack. Example: if the last beat of the correct pattern is a quarter note, the program will accept as correct two eighth notes for the final beat of the user's response.

CHAPTER II

PROGRAM DEVELOPMENT

The purpose of this study was to design and create a series of microcomputer programs that present exercises in rhythmic training for use at the post-secondary level. The skill objectives are two-fold:

1. To correctly associate displayed rhythmic patterns with aurally presented melodies.
2. To recall and notate the rhythmic content of aurally presented melodies.

These objectives are consistent with accepted pedagogical principles appropriate to college level training. Both the material presented and the method of presentation aid in establishing these criteria.

1. Teaching Strategies

Owing to wide variations in the level of ear training skills often present in first year college music students, instructors have to address the needs of lower level students at the expense of the attention and interest of students with better skills. The programs developed in this study provide the instructor with the opportunity to allow students to work at their own pace in achieving a specified level of competency and to free the instructor to aid students having problems.

The programs are designed to be used in conjunction with a first or second year ear training course. They do not contain instructional material covering rhythmic or metric concepts. They function in a tutorial dialogue with the user by presenting exercises in skill mastery and providing immediate feedback in the form of evaluation and diagnosis of the user's response.

The course instructor has responsibilities to present conceptual material, ensuring that students have the prerequisite skills necessary to efficiently utilize the programs, and to establish desired competency levels. The following basic skills are specific conceptual and behavioral objectives prerequisite to the previously stated comprehensive objectives:

1. Knowledge of basic meter types.
2. Knowledge of rhythmic notation.
3. The ability to recognize simple rhythmic patterns presented in non-pitched notation.
4. The ability to articulate simple rhythmic patterns.

Minimum competency and attending grade levels are not established as program parameters; they are left to the discretion of the instructor, providing flexibility for classes of different general skill levels.

2. Program Limitations

The programs developed in this study produce a series of ten lessons; one through five present problems in rhythmic recognition, and six through ten present problems in rhythmic dictation. Each lesson is

designed to be completed in approximately thirty minutes. In actual practice, completion time will vary with the user's skill and aptitude. To establish this projected length, the following criteria were employed:

1. Lessons consist of twenty problems.
2. There is no enforced time limit in completing a problem.
3. Problems may be heard a maximum of three times.
4. Scoring and evaluation are immediate.

Lessons are paired conceptually by the fact that each recognition lesson has a corresponding dictation lesson covering the same melodic material. An enforced completion sequence was not employed in order to allow accommodation of student's varying cognitive styles. Two basic approaches would be to (a) complete all recognition lessons and then complete the dictation series, or (b) complete a given recognition lesson and then complete its corresponding dictation lesson. If desired, an enforced sequence can be added with a minimum of program code.

All lessons are structured in a combined drill-test format. Each time a lesson is attempted or completed, a score is recorded. However, the user has unlimited attempts to achieve the desired score on any given lesson. If the new score is higher than the old, it is automatically stored, if not, it is disregarded. Use of the drill-test format emphasizes and encourages skill mastery, because students are not penalized by a single low test score and can improve their grade by repeating tests.

3. The Musician's Toolbox

The programs developed in this project were written on an "Apple IIe" computer and are capable of running on any computer in the "Apple II" family. Programs are coded in the Applesoft BASIC language and make extensive use of The Musician's Toolbox, hereafter referred to as the toolbox.

The toolbox is a set of 6502 machine language routines authored by Daly, Kolosnick, and Turner (1987). Designed as an aid in the development of music education software, it provides flexible graphics and sound generation capabilities with a minimum requirement of code needed to access its capabilities.

The principal uses of the toolbox in this project are as follows:

1. The combination of graphics and text on the work screen.
2. Ease in manipulating graphics on the work screen.
3. Generation of all sounds used in the programs.
4. The option of using any of six hardware configurations for sound generation.
5. The creation and use of an alternate character set comprised of beamed rhythmic notation symbols.

While full use of these features greatly reduced the time spent in creating portions of the programs, limitations inherent in the toolbox strongly influenced program development and design.

The chief weakness is the large amount of computer memory the toolbox requires for operation. Only a small percentage is available for program use, thus limiting the size of programs that utilize it.

Consequently, the lesson series is produced by a group of separate programs that are linked together rather than by one large program. This limitation is ameliorated considerably by the data storage methods of the toolbox.

4. The Shell Concept

All data used by the programs are not contained in the programs themselves, but are stored on the program disk in the form of binary files. This format allows the programs to be much smaller and to use less of the computer's memory. Programs are "shells" which can manipulate separate data files and, therefore, utilize any file of suitable content and structure. The contents of any file can be easily changed, and new files containing other material can be created without altering the programs themselves, thus providing flexibility in the selection of the material presented. The concept of the shell program is further extended by the recognition lesson series.

All recognition lessons are created by a single program that loads and runs with the appropriate data file, according to which lesson was selected. Owing to the dissimilarity of their graphic designs and other algorithmic elements, each dictation lesson is a separate program. Accordingly, the series of ten lessons is produced by only six programs.

5. Record Keeping and Security

A record keeping system developed at the University of Tennessee was implemented with the lesson series allowing the course instructor

to create a file that contains the course record of each member of the class, and is stored on the program disk. Test scores for each lesson and the number of problems attempted on the recognition and dictation sections of the series are recorded. A separate program allows the instructor to print out the complete class record. This feature serves as a valuable aid to the instructor, not only by considerably reducing the time spent on paperwork, but also by providing concise information on individual student's progress that can be used for a variety of research.

An existing security system was also implemented. To gain access to the lesson series, students first enter their social security number. The program checks this number by comparing it to a list stored on the program disk. The student then enters a given name, and it is compared to the list. If both social security number and name are accepted, the student is allowed into the lesson series. If at either point the student's entry is rejected, the process begins again.

While working on any given lesson, the student's name is displayed on the work screen. This measure was implemented in order to ensure proper allocation of credit for work done by having a lab assistant monitor the work stations and compare the displayed name with the student working. This system will easily disclose any discrepancies.

CHAPTER III

DATA DESIGN

All data used by the lesson programs are stored in five banks containing seventy-five melodies each for a total of 375 melodic rhythms. All melodies in a given bank are written in a single meter. All examples are one measure plus one beat in length and terminate with a quarter or dotted quarter note. Meters were selected on the dual basis of controlling example length and providing for both simple and compound beat division patterns. Meters incorporated are: simple duple, simple triple, simple quadruple, compound duple, and compound triple. Users are not tested on the metric configuration of problems, and the meter signature for each problem is displayed on the work screen.

1. The Basic Rhythmic Pattern Set

Design of the example melodies was predicated by the primary skill objectives. Melodies used by the programs are organized by a set of one beat rhythmic patterns comprised of different division and subdivision schemes.

The basic rhythm pattern set is restricted by the following conventions: simple meter patterns contain one, two, three, or four attacks per beat; compound meter patterns contained one, three, four, or five attacks per beat. Simple syncopated patterns limited to eighth

note syncopations were also used. No rests were included in the pattern set. Figure 1 illustrates all patterns used in construction of the example melodies.

Simple Rhythmic Patterns



Compound Rhythmic Patterns



Syncopated Patterns

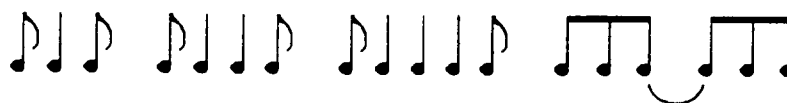


Figure 1. The Basic Rhythm Pattern Set.

The rhythmic content of each example melody is comprised of combinations and permutations from the basic rhythm set. During the course of development of the set, instructors from the University of Tennessee Music Department were consulted, and their recommendations were followed in its creation.

2. Pitch Content of Melodic Examples

Rhythm is the primary organizational element in all example melodies. Pitch sequence has an important but clearly secondary role; it serves primarily as a memory aid for the user and provides a musical context for the rhythmic exercises.

The tessitura of the entire set of melodies encompasses a twelfth, from b to e2, with most individual melodies covering a range of a tenth or less. All melodies are written in the key of C, with the tonalities of C major, F major, a minor, and d minor implied. Owing to the brevity of the melodies, simple cadential harmonies are used exclusively. Figure 2 shows typical harmonic configurations.

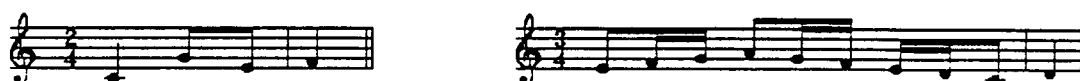


Figure 2. Typical Harmonic Configurations.

The restrictions on pitch sequence were much more prominent in melodic design than its contributions. Two basic conventions whose requirements are dissimilar and at times antagonistic were utilized in constructing the melodies.

The first convention is implicit in the use of The Musician's Toolbox for sound generation software. It lacks the capability for a metric accent; all attacks are of equal weight. Therefore, in order to

enhance the perception of an implied metric accent, pitch sequence must reinforce meter in each example. This is accomplished by two means: pitch placement and agogic accent.

Pitch placement refers to the location of chord tones in the pitch sequence. Locations are determined by the hierarchical relationships of beats and beat divisions implied by the meter of each example. Chord tones are almost always a strong beat or strong portion of a beat. The only exceptions are certain syncopated examples.

Agogic accent is defined as the stress a pitch receives through duration, not through dynamic or metric accent. In the melodic examples, chord tones are almost always longer than nonchord tones, and they have a strong metric placement. Again, syncopated examples are occasional exceptions.

The second convention is occasioned by the problem structure of the recognition lesson programs. Each problem consists of three rhythmic patterns that are displayed, only one of which is heard. After responding and grading, the user has the option of playing any or all of the three examples. To avoid possible confusion, each melody in a group of three must have the same pitch sequence. This assures a logical similarity, should the user compare the example selected to another and forces the user to discriminate on the basis of rhythm to detect any variation.

3. Rhythmic Content of Melodic Examples

The tripartite grouping of examples in order to accommodate the recognition problem lesson structure resulted in certain constraints

being placed on the rhythmic component of each melody. Each melody in a group must serve as a logical foil for the other two by virtue of its similarity. At the same time, the rhythm of each example must be separate and distinct. To accomplish this, the following restriction was applied.

Each melody in a given tripartite group has the same total number of attacks. This prevents the user from counting the attacks heard and using that information as a possible basis for judgement. All groups follow this convention with the exception of a small number which contain syncopated examples.

Three methods were used in combining patterns from the basic rhythm pattern set to form the example melodies. The primary method utilized was to assign all examples in a group the same number of attacks on any given beat and to achieve variation by altering the division or subdivision pattern on selected beats. This provides maximum similarity among examples and enables precise control of differentiation. Figure 3 illustrates this method.

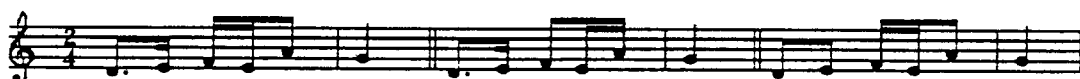


Figure 3. The Primary Method of Pattern Combination.

Use of the primary method forces the user to concentrate on the specific sequence of beat patterns in each example in both recognition and dictation lessons. Approximately 59 percent of the total number of examples were created in this way.

Two other methods were used in combining patterns. Of the remaining examples, 33 percent were written by applying the primary method to a portion of the beats in an example and allowing the remaining beats to have both a different number of attacks and a separate division pattern. Figure 4 illustrates this method.

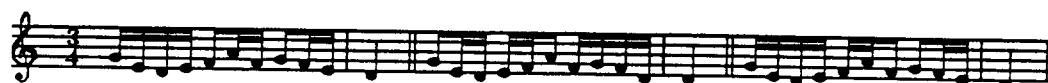


Figure 4. The Secondary Method of Pattern Combination.

Syncopated examples comprise the remaining 9 percent of the total number. To provide differentiation in the tripartite groups that contain syncopation, syncopated elements were placed with examples of an equal number of attacks or with examples where the syncopated element had a different metric placement. Figure 5 illustrates these methods.



Figure 5. Syncopated Combinations.

CHAPTER IV

PROGRAM DESIGN

1. Control of Program Flow

Control of the lesson series user sequence is established by a menu based format for lesson selection. A central menu program controls access to and from each lesson and exit from the lesson series. It operates in the following manner.

A depiction of recognition and dictation lessons and the score recorded for each is displayed on the menu program work screen. The user then selects the desired lesson by pressing its corresponding key (one to ten) and validates this choice by pressing the "Y" key. Pressing the "N" key will invalidate the response and enable the user to make another choice.

The menu program then calls the selected lesson program, supplies information necessary to load the proper data files and runs the program. Once the lesson program is running, the user has options of completing the lesson and automatically returning to the menu or quitting and returning to the menu before the lesson is completed by pressing the "Q" key. The user must exit the lesson series from the menu program.

A series of user help screens is provided as a menu program option. These screens display lists of instructions that detail the complete key-press sequence for each lesson. Because the work screen

of each lesson contains prompts that explain all key-presses, help screens are intended as user aids rather than integral parts of a lesson. Consequently, they are included in the menu program.

2. Screen Design

Design of the work screens for the lesson series was influenced by a variety of factors. Chief among them was the maximum size of the graphics page offered by the toolbox. The graphics page is the area of the computer's video screen available for the combined display of graphics and text. It has a rectangular shape that is forty characters in length and twenty-four characters tall. All visual information presented by the programs is displayed within it. The remaining design factors were strongly influenced by the effort to rearrange the desired elements in this space.

The following information is included in all lessons and is displayed on the work screen of each:

1. The identifying number of the lesson.
2. The user's name.
3. The user's previous high score for that lesson.
4. The user's present score.
5. The number of problems remaining.
6. The number of problem hearings remaining.

Inclusion of these items needs no justification except to remark that the user's previous high score serves as a motivating factor by establishing a goal to be surpassed, thus providing a game-like element to the program.

Prompts are included in each lesson to explain the key-presses necessary to complete the lesson; their number and content vary with the lesson in question. Certain prompts are not on the screen at all times owing to the conflicting requirements of the problem sequence and the space available for display.

Each lesson screen has a problem presentation area where rhythmic patterns are displayed or notated. This area was made large enough to include prompts which identify both the user's response and the answer.

As previously discussed, all five recognition lessons are created by a single program. The size of the problem presentation area is consequently the same for each and was made large enough to display the longest rhythmic patterns (those in compound triple meter). To compensate for various pattern lengths, parameter values passed from the menu program within the presentation area of each lesson.

In dictation lessons, rhythmic characters used to notate problems are displayed on the work screen (in addition to items previously mentioned). The size of the problem presentation area was therefore determined by the space remaining and the length of the patterns presented; it is unique to each lesson.

The discussion thus far has focused primarily on the inclusion and delimitation of graphic elements. Obviously, screen design was not only influenced by these factors but also by a strong need and desire to produce a clear and visually pleasing design that does not distract the user. The latter is subjective and impossible to precisely define. In addressing this concern, instructors from the University of

Tennessee were consulted extensively, and their recommendations are prominent in the end result.

3. The Problem Presentation Sequence

Error Messages

Error messages were used to lock in the lesson key-press sequence. If the user selects an inappropriate response, a prompt momentarily appears, and the selection is disregarded. The following list details conditions that warrant error prompts:

1. Attempts to make any response before listening.
2. More than three listening attempts.
3. Attempts to grade before responding.
4. Ties adjacent or as the first or last character in a user response.

Prompt content is specific to each condition:

1. "You must listen first!"
2. "No listenings left!"
3. "No response to grade!"
4. "Illegal pattern!--Press <E>."

These conditions and prompts restrict the user's key-press options to valid choices and thereby ensure positive control of the user sequence.

The Recognition Lesson Problem Algorithm

Each recognition lesson consists of a series of twenty problems structured in aleatory sequence derived by a two-stage randomizing

process. In the first stage, the lesson data file of twenty-five tripartite groups of melodies is scrambled by means of a random number generator. All groups of melodies are not presented each time a lesson is attempted; only the first twenty from the list are selected in order to avoid the possibility of the user's habituation through repeated exposure. In the second stage, one of the three melodies in each group is randomly selected as the problem stimulus, and thus the twenty problem sequence is formed.

Each problem is structured in the following manner. First, three rhythmic patterns are displayed on the work screen. The user then presses the "L" key in order to hear the stimulus. At this point, any other key-press will result in an error message.

Next, the user can play the example again or enter a response as to which pattern was played, by pressing its corresponding number ("1," "2," or "3"). When a pattern is selected, it is highlighted on the screen. Three hearings are provided and are available at any time before grading. If the user tries to listen more than three times, an error message will momentarily appear. After responding, the user may change the response entered simply by making another choice or may press the "G" key to grade the problem.

Upon grading, if the response was correct, a prompt will so indicate; a message will instruct the user to continue to the next problem. The user's present score is then incremented by five points, and the problem counter is decremented. If the user's response was not correct, a prompt so indicating will appear, the correct and selected choices will be labeled, and the problem counter will be decremented.

After grading, any example may be played by pressing its corresponding key ("1," "2," or "3"). To encourage comparison of the correct and selected response, no restriction was placed, at this point, on the number of times an example could be played. The user has the option of listening and comparing examples or moving to the next problem by pressing the space bar.

When the sequence of twenty problems has been completed, the menu program runs automatically and the user can then work on the same lesson, a new lesson, or exit the lesson series.

The Dictation Lesson Problem Algorithm

The dictation lesson problem algorithm is similar to the one previously discussed in that twenty problems are presented and a two stage randomizing process is employed. However, owing to the presentation of single examples instead of a group of three, its randomizing process has a different result.

In operation the process is identical; the twenty-five groups of the lesson data file are randomized, and the first twenty are selected. One of the three examples in each group is then randomly selected to form the final problem list. The result is that twenty problems are selected at random from a pool of seventy-five available examples. This method was employed because of the similarity of melodies within a group and the attending need to avoid the presentation of examples from the same group in a given lesson.

The problem presentation sequence begins with the user pressing the "L" key to listen to the example. Again, the key-press sequence is

locked in by error messages; three playings may come at any time before grading and an error message will follow a fourth listening attempt.

Next, the student notates the rhythm of the melody that was played, building it up by a two-stage process. First, the user presses the indicated keys (right arrow, left arrow, and "C") to move an arrow pointing to the displayed rhythmic patterns. A separate key press ("N") notates the indicating pattern in the screen work area. The option of erasing the pattern by pressing the "E" key is available to the user at any point before grading.

The user then presses the "G" key to grade the problem. If the response was correct, prompts so indicating appear along with a message to continue to the next problem. The present score is then incremented five points, and the problem counter is decremented. If the response was incorrect, prompts so indicating appear followed by the correct rhythm. The problem counter is decremented, the correct and incorrect rhythms are displayed vertically adjacent, and the discrepancies in the student's response are highlighted.

At this point, the user can play either rhythm by pressing "C" (correct response) or "Y" (your response), or continue to the next problem by pressing the space bar. Once again, no restrictions were placed on the number of playings.

As before, at the end of the lesson the menu program runs automatically.

4. The Dictation Lesson Playback Algorithm

In the previous subsection, it is stated that after grading the user has options of playing either the answer rhythm or the response rhythm. When the answer rhythm is selected for listening, the stored melody from the example pool is played. However, when the response rhythm is selected for listening, the playback algorithm creates a melody by extracting the pitch sequence from the correct melody and combines it with the rhythmic pattern notated by the user. This process is referred to as pitch mapping.

The pitch mapping process begins with the extraction of the pitch sequence of the answer melody. All melodies used by the lesson series are stored on the program disk in the form of lists of numerical data; each note in a melody is represented by a rhythm number and a pitch number. To extract the correct pitch sequence, the playback algorithm selects and orders the numbers that represent the individual pitches of the answer melody and stores them as a list.

Next, the rhythmic patterns of the answer and the response are analyzed for the number of attacks on each beat by comparing them to a stored table of rhythmic values. The playback algorithm then adjusts the pitch sequence in the following manner.

The stored pitch sequence is divided into section according to the number of attacks per beat contained in the answer melody. For example, if the first beat of the answer melody has three attacks, then the first melodic segment equals the first three notes of the melody. If the second beat has two attacks, then the second melodic segment

equals the fourth and fifth notes of the melody. This process is continued in similar fashion to the end of the melody. These segments are then combined with the rhythmic values of the response pattern on a beat by beat basis, adding or subtracting notes as needed.

The length of each melodic segment is then compared to the number of attacks on the corresponding beat of the response rhythm, and a new melody is created on this basis. If on the beat in question the response rhythm has fewer attacks than the answer rhythm, the program simply truncates that melodic segment and continues to the next segment. If the response rhythm has an equal number of attacks on the beat in question, the segment is not altered. If it has more attacks on the beat in question, the program "wraps around" or returns to the beginning of that particular melodic segment and adds notes from that point. Figure 6 illustrates the truncation and extension functions of the playback algorithm. Example one is the original melody. Example two is the altered melody, truncated on the first beat and extended on the second beat.



Example 1.

Example 2.

Figure 6. The Truncation and Extension Functions of the Playback Algorithm.

Next, the new melodic segments are combined with the rhythmic values of the response rhythm, resulting in a new melody that has the approximate pitch sequence and contour of the answer melody and the rhythmic content of the response pattern. The new melody is stored and is then ready to be played.

This algorithm was designed to provide the user with an acceptable approximation of the problem stimulus melody when comparing it to the response rhythm. The design objectives entailed are as follows:

1. In typical errors, the new melody has the same pitch sequence as the old.
2. In every case, the important tonal pitches on downbeats are retained in the new melody.
3. In every case, the first and last pitches are the same in both melodies.
4. In every case, the melodic contour of the problem stimulus melody is preserved to a recognizable extent.

Alternate methods were considered, but the pitch mapping process was selected as the method that produced the most satisfactory results. While it is impossible to precisely predict its outcome owing to the variable factor of the user's input, the pitch mapping process results in very close approximations over a wide range of conditions.

5. The Alternate Character Set

As discussed earlier, both advantages and disadvantages inherent in the use of The Musician's Toolbox had a strong influence on program design. While the toolbox provides easy access to stored melodies, by

virtue of its playback and display functions, the display function is limited by the use of vocal notation; there is no provision for beamed notes. In order to display rhythmic patterns containing beamed notes and to circumvent certain limitations concerning the display of ties and dotted notes, an alternate character set of twenty-two elements was developed.

The melodic display function of the toolbox could not be altered to make use of the new character set; therefore, a different strategy was devised. The toolbox has a separate function to store and display text messages similar to the melodic storage and display capability. This text display function uses both the existing and alternate character sets to display all rhythmic patterns used in the lesson series.

The alternate character set is comprised of beamed, non-pitched rhythmic patterns, dotted notes, ties, and bar lines. Appendix A illustrates the entire alternate character set. The size and style of all characters was based on existing toolbox characters, because of the combined use of the existing and alternate sets.

A strong effort was made to depict the relative proportions of rhythmic durations. For example, the third sixteenth note of a four-note group is the same distance from the first as the second eighth note of a two-note group. Figure 7 illustrates this example.



Figure 7. The Relative Proportion of Beamed Rhythmic Patterns.

All information used by the computer to display the alternate character set is stored on the program disk in the form of blocks of numerical data. The computer uses this data to form individual characters in the following manner.

The display screen is composed of a grid of very small areas or pixels. Each pixel is capable of being made to fluoresce or "turn on." An individual character is displayed by turning on the pixels needed to form its shape. The numerical data used by the computer represents the relative locations of the pixels needed to form each character.

The data were derived by mapping out each character on a master grid. The master grid is further divided into sections, four units wide and one unit high. The numerical value of each segment is calculated by assigning a value of zero to any unit that is not turned on and a value corresponding to equivalent digits in a binary number system to each unit that is turned on. The resulting values are then converted to hexadecimal notation and entered into the computer in the proper sequence.

6. Utility Programs

Twenty-three programs were written in the development of this project. As stated earlier, the lesson series is produced by six programs, and a seventh, the menu program, provides access to the lessons. The remaining sixteen programs serve two functions:

1. The creation of data used by the lesson series.
2. The validation of that data.

Data Creation Programs

Each of the 375 melodies used in the lesson series is represented in three separate categories of data files. One file category contains the actual melodies played by the lesson series. Another contains the depictions of the rhythmic content of the melodies in beamed, non-itched notation. The final category contains information used to grade the user's response. A total of eleven programs create all data stored in these files.

Data Validation Programs

The remaining five utility programs are used to verify that the data created is accurate and that it is stored in the proper sequence. Data are stored in the following relationship.

There are five banks of data, one for each meter type used by the lesson series. Labeled one through five, each data bank contains three files: (1) a play file, (2) a display file, and (3) an answer file. Each file contains seventy-five examples numbered consecutively.

All examples must have an exact correspondence among the three files. For example, in data bank 3, the fifth example in the play file must be depicted exactly by the fifth example of the display file, and the fifth example of the answer file must provide the correct information necessary for grading.

The lesson series requires data that is 100 percent accurate. Given this fact, plus the number of melodies used (375), the need and usefulness of these programs is readily apparent.

CHAPTER V

SUGGESTIONS FOR IMPLEMENTATION

The lesson series developed for this study is scheduled to be implemented as part of the undergraduate music curriculum at The University of Tennessee, Knoxville, during the winter quarter of 1988. It will form part of the freshman music theory sequence and will be used as described in this paper. The ensuing suggestions are intended to aid instructors in serving their students by helping maximize the use of and benefit derived from these lessons.

1. Record Keeping Additions

While these programs are primarily intended to be used as instructional tools, they can also provide data useful in a variety of research. The following possible additions to the programs will be useful in this regard:

1. A record of the number of problems a student completes for each lesson.
2. A record of the number of sessions for each lesson a student attempts.

2. Data Revision

The data used by the lesson series were developed as detailed in Chapter III of this paper. The following possibilities concern alteration of the data, in order to accommodate an instructor's needs:

1. To prevent the user's habituation during problems, enlarge the example pool by:
 - a. Adding new melodies that use the existing rhythmic pattern set.
 - b. Adding new patterns to the basic rhythm pattern set and creating new melodies using them.
2. Transpose the melodic examples to other keys to increase listener interest. This can be done by program code without reprogramming the data files.
3. Alter any objectionable pitch sequence.

3. Algorithmic Changes

The following suggestions concern changing problem delimitations and teaching strategies, in order to accommodate particular class needs:

1. To increase or decrease the difficulty level of a particular lesson:
 - a. Change the number of listenings allowed in a problem.
 - b. Alter the tempo of the melodic examples.
2. Add an enforced lesson completion sequence referenced to a minimum acceptable score.
3. Add programs presenting instructional material covering musical notation, rhythmic, and metric concepts.

With the exception of the final addition, these suggestions could be implemented in the lesson series with a relatively small amount of program code.

BIBLIOGRAPHY

BIBLIOGRAPHY

- Arenson, Michel A., and Fred T. Hofstetter. "The GUIDO System and the PLATO Project." Music Educators Journal, Vol. 69, No. 5 (January 1983), pp. 46-51.
- Bradshaw, Frank, and Virgil Hicks. The Music Class, Vol. 3. Minneapolis: Wenger Music Learning Division, 1987, 26 pgs. and 1 mini floppy disk.
- Daley, Paul, J. Timothy Kolosnick, and Micheal W. Turner. The Musician's Toolbox. Dubuque, Iowa: William C. Brown Publishers, 1987, 123 pgs. and 1 mini floppy disk.
- Deihl, Ned C. "Computer-assisted Instruction and Instrumental Music: Implications for Teaching and Research." Journal of Research in Music Education, Vol. 19 (1971), pp. 299-306.
- Eddins, John M. "A Brief History of Computer-Assisted Instruction in Music." College Music Symposium, Vol. 21, No. 2 (Fall 1981), pp. 7-14.
- Godfrey, David, and Sharon Sterling. The Elements of CAL: The How-To Book on Computer Aided Learning. Reston, Virginia: Reston Publishing Company, 1982, 286 pgs.
- Gordon, Edwin E. Learning Sequence and Patterns in Music. Chicago: G.I.A. Publications, 1977, 168 pgs.
- Hindemith, Paul. Elementary Training for Musicians. Mainz: Schott, 1946, 237 pgs.
- Hofstetter, Fred T. "Computer-Based Recognition of Perceptual Patterns and Learning Styles in Rhythmic Dictation Exercises." Journal of Research in Music Education, Vol. 29, No. 4 (Winter 1981), pp. 265-278.
- Hofstetter, Fred T. "Microelectronics and Music Education." Music Educators Journal, Vol. 65, No. 8 (April 1979), pp. 38-45.
- Horacek, Leo, and Gerald Lefkoff. Programmed Ear Training, Vol. 2. New York: Harcourt, Brace and World, 1967, 504 pgs.
- Huntington, John F. Computer-Assisted Instruction Using BASIC. Englewood Cliffs, New Jersey: McGraw-Hill, 1979, 229 pgs.
- Kantor, Daniel J., Ed. Coda: The New Music Software Catalog. Owatonna, Minnesota: Wenger Corporation, 1987, 160 pgs.

- Lathrop, Ann, and Bobby Goodson. Courseware in the Classroom: Selecting, Organizing and Using Educational Software. Reading, Massachusetts: Addison-Wesley Publishing Company, 1983, 191 pgs.
- Maddison, John. Education in the Microelectronics Era: A Comprehensive Approach. Norfolk, England: Open University Press, 1983, 200 pgs.
- Minnesota Educational Computing Consortium. Music Theory, Vol. 1. St. Paul, Minnesota: Educational Computing Consortium, 1983, 118 pgs. and 1 mini floppy disk.
- Moorhead, Jan Paul. "Computers and Your Brain: Musical Aerobics." Electronic Musician, Vol. 3, No. 10 (October 1987), pp. 26-36.
- O'Neil, Harold, Ed. Computer-Based Instruction: A State-of-the-Art Assessment. New York: Academic Press, 1981, 260 pgs.
- Parr, Phyllis, David L. Shrader, and David B. Williams. Rhythmic Dictator. Bellevue, Washington: Temporal Acuity Products, 1982, 8 pgs. and 1 mini floppy disk.
- Placek, Robert W. "Design and Trail of a Computer-Assisted Lesson in Rhythm." Journal of Research in Music Education, Vol. 22, No. 1 (Fall 1974), pp. 13-23.
- Placek, Robert W. "Choosing the Best Software for Your Class." Music Educators Journal, Vol. 72, No. 1 (September 1985), pp. 49-53.
- Poole, Lon, Martin McNiff, and Steven Cook. Apple II User's Guide, 3rd Ed. Berkeley, California: McGraw Hill, 1985, 588 pgs.
- Rogers, Michael R. Teaching Approaches in Music Theory: An Overview of Pedagogical Philosophies. Carbondale, Illinois: Southern Illinois University Press, 1984, 223 pgs.
- Rumery, Kenneth R. "Computer Applications in Music Education." I.H.E. Journal, Vol. 14, No. 2 (September 1986), pp. 97-99.
- Schrader, David L. "Microcomputer-Based Teaching: Computer-Assisted Instruction Comes of Age." College Music Symposium, Vol. 21, No. 2 (Fall 1981), pp. 27-36.
- Walker, Decker F., and Robert D. Hess. Instructional Software: Principles and Perspectives for Design and Use. Belmont, California: Wadsworth Publishing Company, 1984, 308 pgs.
- Wolfe, George. "What to Look for in Educational Software." Jazz Educators Journal, Vol. 16, No. 4 (April/May 1983), pp. 23-25.
- Wolfe, George. "Choosing Music Peripherals for CAI." Jazz Educators Journal, Vol. 17, No. 1 (October/November 1984), pp. 27-29.

APPENDIXES

APPENDIX A

THE ALTERNATE CHARACTER SET



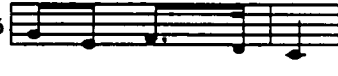
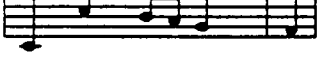
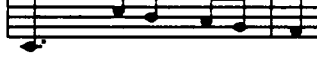
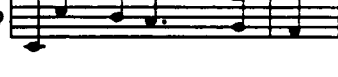
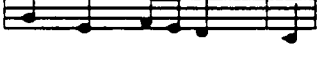
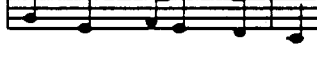
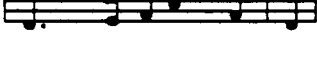
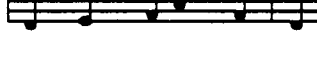
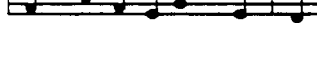
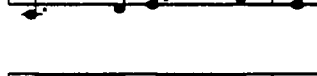
APPENDIX B

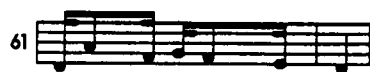
MELODY BANK 1



1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30



31		32		33	
34		35		36	
37		38		39	
40		41		42	
43		44		45	
46		47		48	
49		50		51	
52		53		54	
55		56		57	
58		59		60	



61



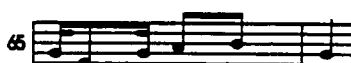
62



63



64



65



66



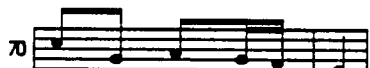
67



68



69



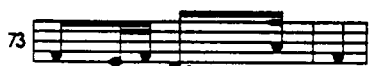
70



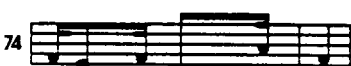
71



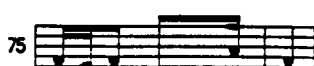
72



73



74



75

APPENDIX C

MELODY BANK 2



1 2 3

4 5 6

7 8 9

10 11 12

13 14 15

16 17 18

19 20 21

22 23 24

25 26 27

28 29 30

A musical score for a single melodic line on a treble clef staff, spanning measures 31 to 60. The notation is in a single system with three staves per row. The melody consists of eighth and sixteenth notes, often beamed together in groups. Measure numbers 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, and 60 are printed at the beginning of each measure. Triplet markings, consisting of a '3' above a bracket, are placed over measures 46, 50, 54, 55, 56, and 58. The key signature is one flat (Bb), and the time signature is 4/4.

Musical score for piano, measures 61-75. The score is written on a single staff with a treble clef. The key signature is one flat (B-flat). The time signature is 4/4. The music consists of a continuous eighth-note pattern with occasional triplet markings (indicated by a '3' over the notes). The measures are numbered 61 through 75.

Measures 61-75:

- 61: Triplet of eighth notes, followed by eighth notes.
- 62: Triplet of eighth notes, followed by eighth notes.
- 63: Triplet of eighth notes, followed by eighth notes.
- 64: Triplet of eighth notes, followed by eighth notes.
- 65: Triplet of eighth notes, followed by eighth notes.
- 66: Triplet of eighth notes, followed by eighth notes.
- 67: Triplet of eighth notes, followed by eighth notes.
- 68: Triplet of eighth notes, followed by eighth notes.
- 69: Triplet of eighth notes, followed by eighth notes.
- 70: Triplet of eighth notes, followed by eighth notes.
- 71: Triplet of eighth notes, followed by eighth notes.
- 72: Triplet of eighth notes, followed by eighth notes.
- 73: Triplet of eighth notes, followed by eighth notes.
- 74: Triplet of eighth notes, followed by eighth notes.
- 75: Triplet of eighth notes, followed by eighth notes.

APPENDIX D

MELODY BANK 3



1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30

31 32 33

34 35 36

37 38 39

40 41 42

43 44 45

46 47 48

49 50 51

52 53 54

55 56 57

58 59 60

3

3

3

3

3

3

3

3

3

3



61 62 63

64 65 66

67 68 69

70 71 72

73 74 75

APPENDIX E

MELODY BANK 4



1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30



The image displays a 6x3 grid of musical staves, numbered 31 to 60. Each staff contains a sequence of musical notes and rests, primarily in a single melodic line. The notation includes eighth notes, quarter notes, and half notes, often grouped with beams. Some staves feature slurs or ties. The staves are arranged in six rows and three columns. The first row contains staves 31, 32, and 33; the second row contains 34, 35, and 36; the third row contains 37, 38, and 39; the fourth row contains 40, 41, and 42; the fifth row contains 43, 44, and 45; and the sixth row contains 46, 47, and 48. The seventh row contains staves 49, 50, and 51; the eighth row contains 52, 53, and 54; the ninth row contains 55, 56, and 57; and the tenth row contains 58, 59, and 60. The musical notation is consistent across all staves, suggesting a single melodic line or a specific exercise.



A musical score for a single melodic line, measures 61 through 75. The notation is on a single staff with a treble clef at the beginning. The key signature has one sharp (F#). The time signature is 4/4. The melody consists of eighth and sixteenth notes, often beamed together in groups of four. Measures 61-63 show a sequence of eighth notes. Measures 64-66 show a sequence of sixteenth notes. Measures 67-69 show a sequence of eighth notes. Measures 70-72 show a sequence of sixteenth notes. Measures 73-75 show a sequence of eighth notes. The score is arranged in five rows of three measures each.

61 62 63

64 65 66

67 68 69

70 71 72

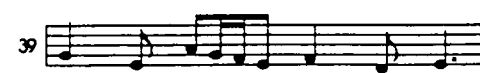
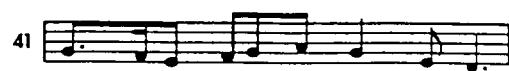
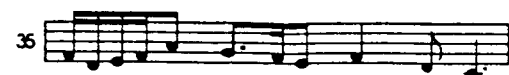
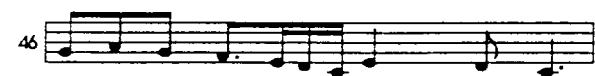
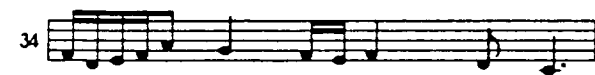
73 74 75

APPENDIX F

MELODY BANK 5



1	2	3
4	5	6
7	8	9
10	11	12
13	14	15
16	17	18
19	20	21
22	23	24
25	26	27
28	29	30





61 62 63

64 65 66

67 68 69

70 71 72

73 74 75

This musical score is written on a grand staff with a treble clef. It consists of 15 measures, numbered 61 through 75. The notation includes various rhythmic values such as eighth, sixteenth, and thirty-second notes, as well as rests. The measures are arranged in five rows of three, with the final row containing measures 73, 74, and 75.

VITA

John Mitchell Barron was born in Thomaston, Georgia on June 14, 1953. He attended elementary school in that city and was graduated from Robert E. Lee High School in June 1971.

After working in the construction business in the southeastern United States for ten years, he entered Columbus College (Columbus, Georgia) in the fall of 1981 and received a Bachelor of Music Education degree in June 1985.

In the fall of 1985 he accepted a teaching assistantship at The University of Tennessee, Knoxville and began study toward a Master of Music degree with a concentration in theory. This degree was awarded in March 1988.