Designer: Stephen R. Shalom

Editors: Tom Prosen
Susan Markey
Marge Kosel

Programmer: Stephen R. Shalom

COPYRIGHT 1984
SUNBURST COMMUNICATIONS, INC.
PLEASANTVILLE, NEW YORK 10570

**Apple II is a registered trademark of
Apple Computer, Inc., Cupertino, California
TI 99/4A is a registered trademark of Texas Instruments, Inc. of
Lubbock, Texas**

For information, or a free microcomputer courseware catalog, write:

SUNBURST COMMUNICATIONS, INC.
39 WASHINGTON AVENUE
PLEASANTVILLE, NEW YORK 10570

**Call toll-free (800) 431-1934
(In New York, Alaska, Hawaii, or Canada call collect (914) 769-5030.)**

SPECTRUM: PATTERNS AND PROGRAMS

TABLE OF CONTENTS

INTRODUCTION

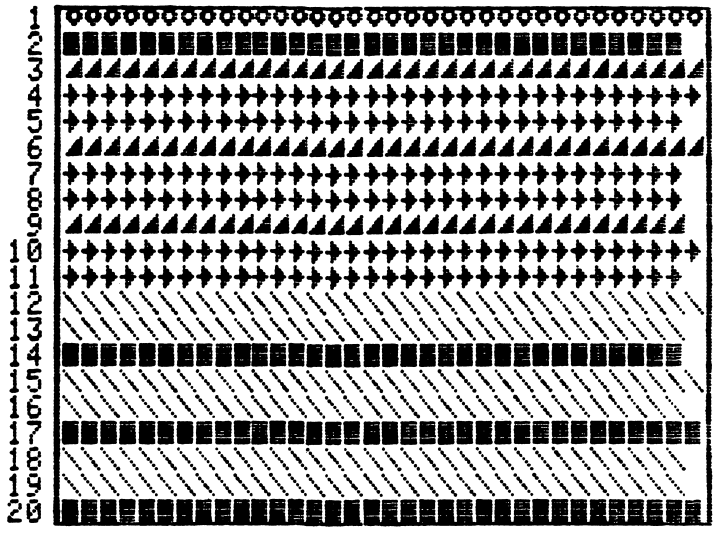
SPECTRUM: PATTERNS AND PROGRAMS is a logic game designed to give players practice in pattern recognition, prediction, logic and inductive reasoning.

The object of the game is to determine a hidden pattern of bars of different colors or shapes with as few clues as possible. To discover a pattern, players obtain clues by requesting the color of a particular numbered bar. Players continue to request clues until they think they have abstracted enough information to make an intuitive hunch about the pattern. Thus, SPECTRUM requires experimentation, note taking, hypotheses, formulation, and the making of predictions based on information acquired.

The "hidden pattern" is generated by a program that may be written by the computer or a player. Once the hidden pattern is recognized, players try to duplicate the pattern by entering their own pattern program. Here, more learning takes place as players build skills by following the logic of computer programs and by writing their own simple programs.

Another step in the learning process can be unveiled by having the computer write the players' programs more efficiently than the players. This provides immediate feedback to their own attempts at writing programs and gives them useful instruction for improving their thinking strategies.

SPECTRUM is designed for the TI/99-4A and a 48K Apple II. If you need assistance in operating your computer, check the sections of this guide entitled "TI/99-4A: Working With the Computer" or "Apple II: Working With the Computer" and "What Happens If...?" or call Sunburst Communications toll-free at 800-431-1934.



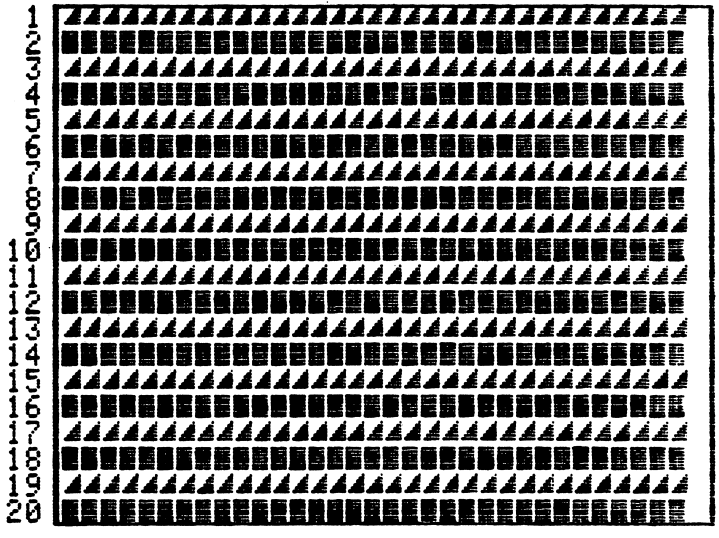
Skills:	Pattern recoginition Prediction Logic Induction Strategy
Students Work With:	Hidden patterns Colors and shapes Sequence
Grade Level:	6-adult
Reading Level:	6th grade (Fry)
Time Required:	5-20 minutes per level
Objectives:	1. To find order and discover patterns. 2. To practice following the logic of simple programs. 3. To build skills in writing simple computer programs.

PROGRAM DESCRIPTION

Game Objective

The object of the game is to discover a "hidden pattern" that has been generated by a program.

The pattern will consist of a number of colored bars (using a colored TV) or a number of shape bars (not using a colored TV). The pattern will be formed through repetition of the bars. A simple pattern of alternating bars of triangles and bars of boxes is shown below:



A score will be awarded which will be partially dependent on the number of clues needed to guess the pattern.

Colors or Shapes

You will be asked whether or not you have a color monitor or television connected to your computer. If you don't, you can still play SPECTRUM, but instead of the patterns consisting of bars of different colors, the bars will consist of different shapes. Of course, if you wish, you can also play with shapes on a color TV. Below are the colors and shapes available:

On the Apple

Colors		Shapes	
green	violet	box	arrow
white	orange	slant	triangle
aqua	black	heart	circle

On the TI

Colors		Shapes	
green	violet	box	arrow
white	red	slant	triangle
blue	yellow	heart	diamond

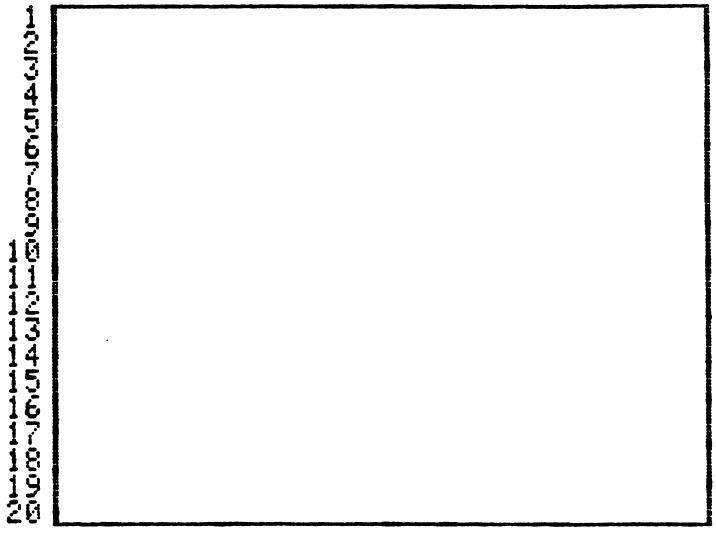
PLAY A GAME

Getting Started

The easiest way to begin is by choosing to "Play a Game" rather than "Create a Pattern". The computer will generate the "hidden program" but the number of players (1-4) and a difficulty level must be chosen. In the nine difficulty levels all colors (shapes) may be used and the patterns always contain 20 bars except for the pattern generated at level 9 which contains 40 bars. (On the Apple, one or more of the difficulty levels may be restricted from student use as described in the Teacher Option.) It is advisable to have beginners start at level one or two.

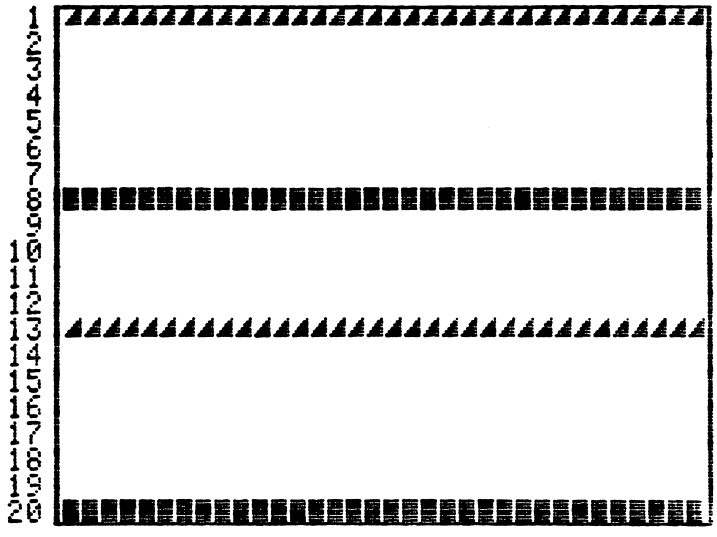
Figuring Out The Patterns

Play begins with the screen showing bars numbered from 1 to 20 (or 1 to 40 in level 9). The initial state of the bars is a white line with a hole in it when playing in color and an empty space when playing in black-and-white (on the TI: a solid gray bar is used).



Notice the lower portion of the screen has the bar request display. This display has a space where the player can choose the number of a bar they wish to reveal. Players try to figure out the pattern by asking for clues. To see the clue (the color or shape of a bar), the player simply types in its number and presses the RETURN key (on the TI: the ENTER key). The indicated bar will change from its initial state to its shape (color) in the hidden pattern. The player tries to reveal bar shapes (colors) in such a way

that the pattern can be deduced. There is no limit, short of the entire pattern, to the number of bars that may be requested, but the fewer clues needed the higher the player's score will be (see the Scoring section). If all the bars are requested, the round ends. Below is a sample screen after requesting to see bars 1, 8, 13 and 20 from level 2.



Enter # of bar you wish to see:

P = Program L = List ? = Give up

Players who think they know the whole pattern may try to match it. (Trying to match the pattern may also be used as a means of obtaining more clues about the pattern -- see the section on Strategy on page 24.) Matching can be done in one of two ways:

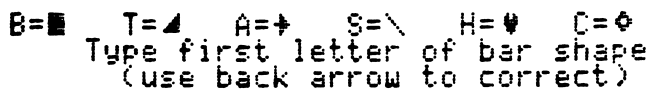
1. L = List bar-by-bar listing of colors (shapes)

2. P = Program writing a pattern-generating program

Using Bar-by-Bar Listing

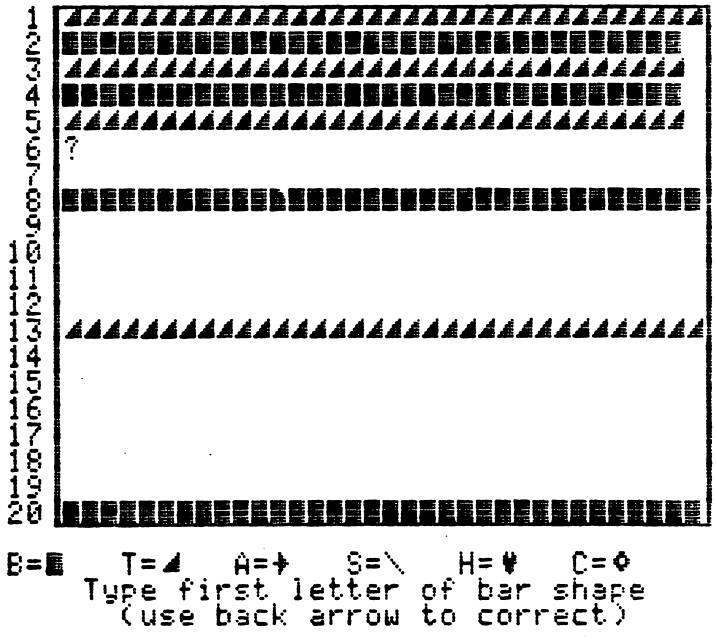
For beginners, the easiest method of matching the "hidden pattern" is by pressing the "L" key (for "List"). (On the Apple: "L" will not appear as a possibility if the Teacher Option has prohibited it. See page 21.) The

shapes (colors) are displayed at the bottom of the screen along with their initial letters as shown below:



A question mark appears in the lowest numbered bar that is still in its initial state (has not been revealed). The player is instructed to type the first letter of the shape (color) that he or she thinks will match the shape (color) with the question mark. The bar changes to the shape (color) selected by the player whether it is correct or not. To distinguish the previously revealed bars from the ones chosen, the latter are shorter. The questions mark then moves to the next unrevealed bar and the procedure is repeated until all the bars are filled-in. The screen below shows bars 1, 8, 13 and 20 previously revealed and then 2, 3, 4, 5 as they would appear by using the List Option.

Pressing the BACK-ARROW key anytime before the final bar has been reached will permit the player to re-enter the previously entered bar. Pressing the BACK-ARROW key when the question mark at the first unrevealed bar returns the player to the bar request display without any penalty.



When all of the unrevealed bars are "Listed" the player is given the:

NUMBER RIGHT -- the number of bars chosen (using the List option) whose shape (color) matches the shape (color) of the bars in the hidden pattern.

NUMBER WRONG -- the number of bars whose shape (color) does not match the shape (color) of the bars in the hidden pattern.

If the number wrong is zero, then the computer will print YOU GOT IT! and the player's score.

If the answer is incorrect, then the computer will display the number of "tries." (Each unsuccessful attempt to match the hidden program counts as a "try".) The bar request display is then returned to the bottom of the screen and the bars that were colored by the bar-by-bar listing are changed back to their initial state. Those that were previously made visible by bar requests, however, are left unchanged (visible). At this time, players may wish to record this information onto a copy of one of the worksheets provided at the back of this guide.

Players may make as many tries as they want, even though each try reduces the score. The players may continue requesting bars and trying to match the pattern until either:

1) the pattern has been matched,

2) all the bars have been filled-in by bar requests,

3) they give up. To "give up," players should enter a question mark (on the TI hold down the FCTN key and press the "I" key). When you "give up," the computer will reveal the colors (shapes) of the remaining bars. No score will be reported.

After the pattern has been revealed the computer will print out the hidden program that generated the pattern.

If there is more than one player, at the end of the game each of the player's scores will be displayed with the winner (or TIE, in the case of a tie) indicated.

Writing a Program-Generating Program

Pressing the "P" key (for "Program") allows the player to try to match the "hidden program" by entering a pattern-generating program of his or her own. In order to write a program, some rules must be clarified. There are two types of program statements in SPECTRUM. The first is a Color (Shape) Statement and the other is the Loop Statement. Some of the examples in this guide will use color statements and others will use Shape Statements. The principles are exactly the same in either case.

The hidden pattern is generated by a program consisting of a series of

statements. As shown below, a program is a set of instructions for the computer to carry out:

1	Triangle	8	Box	15	Triangle
2	Box	9	Triangle	16	Box
3	Triangle	10	Box	17	Triangle
4	Box	11	Triangle	18	Box
5	Triangle	12	Box	19	Triangle
6	Box	13	Triangle	20	Box
7	Triangle	14	Box		

This program instructs the computer to display the pattern shown on page 3. A statement is an individual instruction to the computer. Each statement must begin with a line number. The line numbers tell the computer the order in which to perform the instructions. In SPECTRUM, the line numbers are displayed by the computer beginning with the number one.

Color or Shape Statements

A Color Statement is simply the name of any of the allowed colors. A Shape Statement is simply the name of any of the allowed shapes. Shapes and colors may not both be used in the same game. Players may enter the entire name of the color (shape) or they may enter the first letter (for example, G for GREEN and W for WHITE or A for ARROW and B for BOX).

Loop Statements

Loop statements are statements that cause a group of the instructions to be repeated a specified number of times. Loop Statements are made up of four components:

- the letter "F" (which stands for the word "from")
- the line number from which the loop (repetition) begins
- an asterisk (*) (the computer symbol for multiplication)
- and the number of repetitions to be performed

The four parts are then put together on one line; for example:

```
F1*3
```

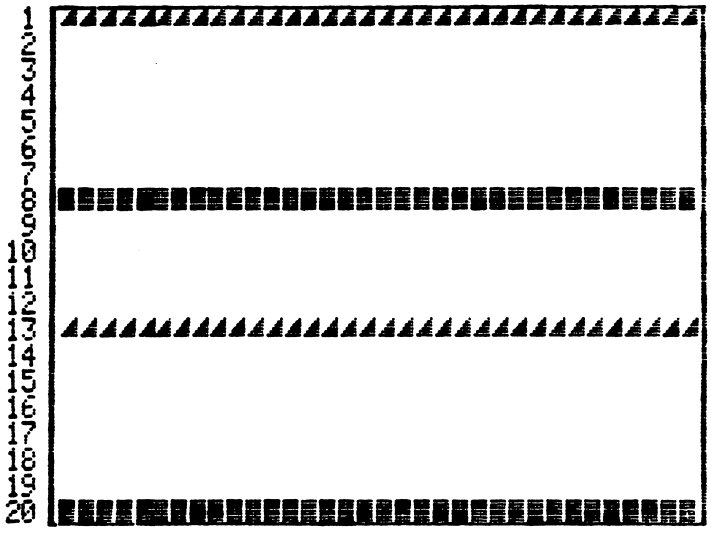
means to begin from line 1 and repeat three times.

In the example on page 3, a 20 line program could be entered with triangle and box alternating up to 20 (as shown at the top of this page). The following is another way to enter the program, but this time using a Loop Statement:

```
1 TRIANGLE
2 BOX
3 F1*9
```

Notice that the computer is told on line one to place a bar of triangles, on line two to place a bar of boxes and when the computer gets to line three it is to repeat starting at line one up to the repeat statement. This is done nine times. Note the nine repetitions means that statements one and two will occur 10 times . . . the initial occurrence and then nine repetitions. More details concerning Loop Statements can be found on page 12 in the "Create a Pattern" section.

To enter the program, the shapes (colors) names are displayed. The player simply types in the appropriate statement following the displayed line number and presses the RETURN key (on the TI: the ENTER key). As shown below only the current line being entered remains on the screen, so it is important for the players to keep notes on paper of what they have entered. Players entering a program must remember that the program must represent the entire pattern not just the unrevealed bars.



Corrections may be made by using the BACK-ARROW key. (On the Apple: pressing the BACK-ARROW key at the beginning of any statement will bring up the previous statement.) Pressing the BACK-ARROW key at the beginning of statement 1 returns to the bar request display without penalty. As each statement is entered, the computer checks that it is error free. If it contains an error, it is rejected and the player is allowed to re-enter the statement.

After the program is entered, the bars are turned from their present state to the shape (color) generated by the matching program. To distinguish the previously revealed bars from the newly filled-in ones, the latter are displayed slightly shorter.

The computer will then print out the following information:

NUMBER RIGHT -- the number of new bars whose shape (color) matches that of the bars in the "hidden pattern."

NUMBER WRONG -- the number of new bars whose shape (color) does not match that of the bars in the "hidden pattern."

INCONSISTENCIES -- the number of bars for which the matching program doesn't match the previously visible bars of the "hidden pattern."

If the number wrong is zero and there are no inconsistencies, then the computer will print YOU GOT IT! and the score. Note the matching program need not be the same as the one that generated the hidden pattern or have the same number of statements; it must merely generate the same pattern. As will be seen in the section on Scoring, the player who entered the hidden pattern will be scored based on the number of statements in his or her program.

If the number wrong or the number of inconsistencies is not zero, then the computer will display the number of "tries." (Each unsuccessful attempt to match the hidden program, whether by writing a program or by bar-by-bar listing, counts as a "try".) The BAR REQUEST DISPLAY is then returned to the bottom of the screen and the bars that were colored by the matching program are changed back to their initial state. Those that were previously made visible by bar requests, however, are left unchanged (visible). At this time, players may wish to record this information onto a copy of one of the worksheets provided at the back of this guide.

After the pattern has been revealed the computer will print out the hidden program that generated the pattern. If the program was entered by a player (see "Creating a Pattern"), the computer will print out a concise (the shortest) program if requested by the player.

If there is more than one player, at the end of the game each score will be displayed with the winner (or TIE, in the case of a tie) indicated.

Details on Difficulty Levels

In general, each of the difficulty levels may produce a pattern from an easier level:

Level 1 - a pattern consisting of either (a) a single color (shape) repeated for all 20 bars, or (b) two colors that alternately repeat 10 times to fill the 20 bars.

Level 2 - a pattern of 1, 2, or 4 bars that are repeated throughout the 20 bars.

Level 3 - a pattern of 1, 2, 4, or 5 bars that are repeated throughout the 20 bars. Note that within the 4th or 5th bar pattern there may be additional repetitions for example, GVGVW GVGVW GVGVW GVGVW generated by the program:

```
1  G
2  V
3  F1*1
4  W
5  F1*3
```

Level 4 - two patterns, each consisting of some repetitions of 1, 2, or 3 bars. Since the patterns are generated randomly within the constraints of the level, it is possible that the two patterns will be the same and then there will be, in effect, just a single pattern.

Level 5 - two patterns, each repeating from 1 to 6 bars.

Level 6 - same as level 5 or else will contain a single pattern from 1 to 3 bars and 2 random bars at the beginning or end.

Level 7 - two patterns of 1 to 6 bars with random bars possible at the beginning, end, or in between.

Level 8 - no restriction on the computer: it will simply generate a pattern at random for 20 bars; it may be as regular as 20 green bars or as irregular as 20 bars with no repetition at all (anything goes).

Level 9 - "anything goes", but this time for a pattern of 40 bars.

CREATING A PATTERN

Initial Conditions

If the player chooses to write a program to create the "hidden pattern", a number of initial conditions (parameters) must first be set:

1. The number of players (1-4) must be chosen. When there are two, three, or four players, the first player enters a program which will generate a pattern for the second player to figure out, then the second player enters a program to challenge the third, etc. Finally the last player enters a program to challenge the first. All parameters set by the players remain the same for each of the players.

2. If there is more than one player, the players have the option of entering their names. Those who do not choose to enter their names will be identified by the computer as Player 1, Player 2, etc.

3. The number of bars. Programs can be written to generate patterns for 20 bars or for 40 bars.

4. The number of permissible colors (or shapes) may be set at 3, 4, 5, or 6. Not all permissible colors need to be used in writing a program; the number selected constitutes an upper limit of which colors may be used. A list of all of the available colors and shapes follows:

On the Apple

Colors		Shapes	
green	violet	box	arrow
white	orange	slant	triangle
aqua	black	heart	circle

On the TI

Colors		Shapes	
green	violet	box	arrow
white	red	slant	triangle
blue	yellow	heart	diamond

Some of the examples in this section of the guide will use colored bars and others will use shapes. The principles are exactly the same in either case.

The pattern is generated by a program consisting of a series of statements. As shown below a program is a set of instructions for the computer to carry out:

1	Triangle	8	Box	15	Triangle
2	Box	9	Triangle	16	Box
3	Triangle	10	Box	17	Triangle
4	Box	11	Triangle	18	Box
5	Triangle	12	Box	19	Triangle
6	Box	13	Triangle	20	Box
7	Triangle	14	Box		

This program instructs the computer to display the pattern shown on page 5. A statement is an individual instruction to the computer (20 HEART). Each statement must begin with a line number. The line numbers tell the computer the order in which to perform the instructions. In SPECTRUM, the line numbers are displayed by the computer beginning with the number one.

Color or Shape Statements

A color statement is simply the name of any of the allowed colors. A shape statement is simply the name of any of the allowed shapes. Shapes and colors may not both be used in the same game. Players may enter the entire name of the color (shape) or they may enter the first letter (for example, G for GREEN and W for WHITE or A for ARROW and B for BOX).

Loop Statements

Loop statements are statements that cause a group of the instructions to be repeated a specified number of times. Loop statements are made up of four components:

- the letter "F" (which stands for the word "from")
- the line number form which the loop (repetition) begins
- an asterisk (*) (the computer symbol for multiplication)
- and the number of repetitions to be performed.

The four parts are put together on one line, for example:

```
F1*3
```

This means to begin from line 1 and repeat 3 times. In other words, the loop will repeat for the appropriate number of repetitions from the designated line number to the line number just prior to the loop statement. Study the example below:

```
                    1  BOX
                    2  ARROW
                    3  SLANT
                    4  ARROW
                    5  HEART
                    6  HEART
Loop statement ---- 7  F3*2
```

When the computer gets to line number 7, it will repeat the statements at line numbers 3, 4, 5, and 6 twice. Note that two repetitions means that statements 3 through 6 will occur three times -- their initial occurrence plus the two repetitions. This is an important point to stress with your students.

Loop Statement Rules

The line number following the "F" must be a lower number than the line number of the loop statement itself. Therefore, the first statement of a program may not be a loop.

The number of repetitions may be 1 or any higher whole number so long as the total number of statements implied in the loop plus in the preceding statements does not exceed the total number of bars allowed in the pattern. For example, if the 20-bar version of the game is being played and the first four statements are:

```
1  WHITE
2  GREEN
3  WHITE
4  VIOLET
```

The fifth statement may be any color or a loop. If it is a loop and it repeats from statement 1, it may have as many as four repetitions; more than four repetitions would imply more than 20 bars:

Loop Statement	Total Bars
5 F1*4	4 + (4 x 4) = 20

If the fifth statement repeats from statement 2, there may be from one to five repetitions:

Loop Statement	Total Bars
5 F2*5	4 + (3 x 5) = 19

In this case another bar would need to be added to make 20 bars.

If it repeats from statement 3, there may be up to 8 repetitions:

Loop Statement	Total Bars
5 F3*8	4 + (2 x 8) = 20

From statement 4, up to 16 repetitions may occur:

Loop Statement	Total Bars
5 F4*16	4 + 1 x 16 = 20

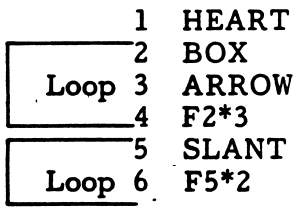
Nested Loop Rules

A "nested" loop is a loop embedded within another loop. In SPECTRUM, loops may be nested, but not "crossed." For example, with the following statements:

```
1  HEART
2  BOX
3  ARROW
4  F2*3
5  SLANT
```

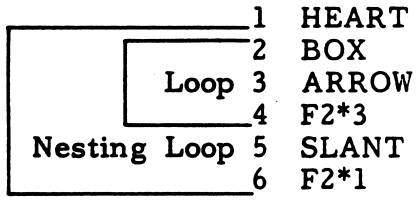
line number 6 could contain a statement that repeated from line 5.

In this case, there would be no nested loop:



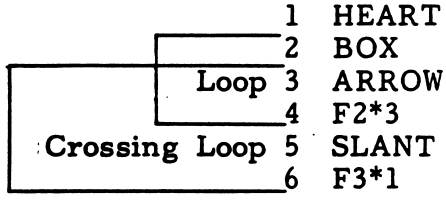
The above pattern would be: HEART, BOX, ARROW, BOX, ARROW, BOX, ARROW, BOX, ARROW, SLANT, SLANT, SLANT, and more statements would be needed to reach 20.

On the other hand, line 6 could contain a statement that repeated from lines 1 or 2. If this strategy was chosen, it would "nest" the loop consisting of statements 2 through 4:



Thus, the above program would be: HEART, BOX, ARROW, BOX, ARROW, BOX, ARROW, BOX, ARROW, SLANT, BOX, ARROW, BOX, ARROW, BOX, ARROW, SLANT, and more statements would be needed to reach 20 but one more loop like statement 6 would be too much.

However, line 6 may not contain a statement that repeats from statements 3 or 4 because it would define a loop that crossed the 2 through 4 loop:



When you ran your program, you would get this: HEART, BOX, ARROW, BOX, ARROW, BOX, ARROW, BOX, ARROW, SLANT...and the computer would want to try to go to line 3 as instructed, but it cannot. A loop cannot start in the middle of another loop. When you type in line 6 the computer will check to see if it is a valid statement and if it's not, as in this case, you will need to re-enter a valid statement.

How to Enter Programs

When a "hidden program" is to be written by a player rather than the computer, the player is first shown a screen indicating the allowable statements as shown below:

```
                ALLOWABLE STATEMENTS

                      B OR BOX
                      T OR TRIANGLE
                      A OR ARROW
                      S OR SLANT
                      H OR HEART
                      C OR CIRCLE

          FL*N --where L is a line number and N
                 is the number of repetitions.


                  Press RETURN to continue.
```

Line numbers are then displayed, starting at 1, and the player may type in any allowable statement. The RETURN (on the TI: ENTER) key must be pressed after each statement is typed. Before the key is pressed corrections may be made by using the BACK-ARROW key. (On the Apple, pressing the BACK-ARROW key at the beginning of a program statement will return you to the previous statement. Typing a "?" and pressing RETURN will redisplay the screen which lists the allowable statements.)

Writing Concise Programs

The shortest pattern-generating program that can be written is two statements long: first a color statement and then a loop statement with the appropriate number of repetitions:

```
1 GREEN
2 F1*19
```

The longest possible program will have as many statements as there are bars. Note that loop statements will always shorten the length of the program except for loops that repeat just the previous statement a single time.

Programs written by players need not be written in the most concise way. The computer will accept, for example:

```
1 VIOLET                                  1 VIOLET
2 VIOLET  instead of the more concise     2 F1*19
3 F1*9
```

or

```
1 BOX
2 HEART                                   1 BOX
3 BOX     instead of the shorter          2 HEART
4 HEART                                   3 F1*9
5 F1*4
```

After a pattern has been created play continues as in "Play a Game" (see pages 4 through 11). Inefficient programs penalize the one who wrote the hidden program and players are given the option, at the end of the round, to see if the computer can write the program for the "hidden pattern" more concisely.

If there is only one player writing his or her own program then there are two options after writing the program:

1. The computer can be called on to write a more concise program.

2. You can still challenge a friend, play then continues.

SCORING

A player's score is computed as follows: the number of program statements in the pattern-generating program that produced the hidden pattern is added to the number of bars correctly matched by the player, and the number of unsuccessful matching attempts is then subtracted. The resulting number is then taken as a percent of the number of bars (20 or 40). In the 20 bar game, therefore, each try costs five points, each statement in the hidden program is worth five points, and each correctly matched bar is worth five points.

The implications of this scoring method are:

1. The more bars you figure out correctly the higher your score will be.

2. The more statements there were in the hidden program the higher your score will be (note: you get more points when the "hidden program" that you matched is longer or less concise than necessary).

3. The fewer attempts you needed at matching the higher your score will be.

There is no lowest possible score since there is no limit to the number of unsuccessful tries. The highest possible score, theoretically speaking, is 200 (20 statements + 20 right * 5 and there were no unsuccessful tries) but as a practical matter, at the higher difficulty levels scores will rarely exceed 100.

If all the bars have been colored in requesting clues, the score is computed in the same way except that the NUMBER RIGHT is zero. For example, if someone were to write a program that requires 18 statements to generate the pattern, then the player who is trying to figure it out could get 90 points (18 + 0 x 5) even though not a single bar has been figured out by the player whose turn it is.

Notice that requesting all the bars is not the same as giving up. When a player gives up, by pressing the questions mark, no score is reported. On the other hand, when a player asks to see all the bars -- as might happen when working on a pattern that is essentially random -- the NUMBER RIGHT is zero, but the number of statements in the hidden program still count toward the player's final score.

ENDING THE GAME

The game may be interrupted at any time:

On the Apple -- by holding down the CONTROL (CTRL) key while pressing the E key. A menu will then asks you if you want to play again. If you would like to play again, enter 1 for yes; otherwise enter a 2 for no to exit the program.

On the TI -- by holding down the Function key while pressing the 4 key. To play again type RUN.

Teacher Options

When the main menu appears (figure 1), holding down the Control key and pressing the letter "T" at the same time will display the Teacher Option menu (figure 2). The SPECTRUM diskette must be in the drive when using the Teacher Options and the write-protect sticker must be removed.

```
Please make choice:

  1. Play a Game

  2. Create a Pattern

  3. Instructions

  4. End

Type number and press RETURN: ■
```

figure 1

```
Teacher Options:
  1. (Un)restrict difficulty level
  2. (Un)restrict bar-by-bar listing
  3. See saved program
  4. Save program
  5. Delete saved program
  6. (De)select saved program
  7. Turn sound on/off
  8. Return to main program

Type number and press RETURN: ■
```

figure 2

NOTE: Be sure to exit the Teacher Options either by selecting the "Return to Main Menu" option or by typing Control-E. If you turn off the computer in the middle of the Teacher Option, the teacher commands may not be properly saved on the diskette.

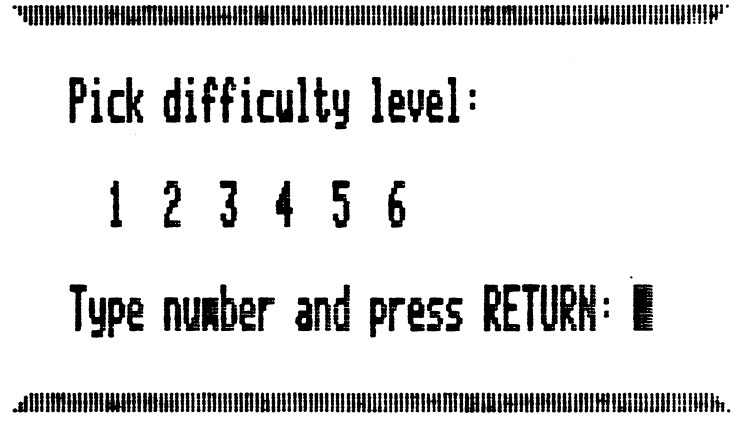
The eight options displayed on the Teacher Options menu are:

1. (Un)Restrict Difficulty Level -- the teacher can control which levels of difficulty are used by the players. Selecting this option displays the status (restricted or unrestricted) of each level as shown below:

```
Difficulty Level        Status

      1                 unrestricted
      2                 unrestricted
      3                 unrestricted
      4                 unrestricted
      5                 unrestricted
      6                 unrestricted
      7                 unrestricted
      8                 unrestricted
      9                 unrestricted

Type number of difficulty level to
change status or type '0' to exit.


Type number and press RETURN: ■
```

By typing the number of a level, the status of that level is changed. Typing a zero "0" returns you to the menu of Teacher Options. As many as 8 of the nine levels may be restricted at one time. Restrictions are recorded on the diskette and will remain in effect even if the computer were turned off and then used at another time.

When SPECTRUM is run for the first time, all of the 9 difficulty levels are unrestricted, that is, they are all available to the player. When a level is restricted, it is unavailable to the player and the menu displaying the choice of the difficulty levels for the player will not display the numbers of the restricted levels. Figure 4 shows choice a screen with levels 7, 8, and 9 restricted is shown below. Recall that difficulty levels only refer to pattern-generating programs written by the computer; they have no bearing on programs written by players.



2. (Un)Restrict Bar-By-Bar Listing -- allows the teacher to prohibit students from trying to match the hidden program by bar-by-bar listing. Players will only be able to try to match the program by writing a program of their own. When SPECTRUM is first used, both bar-by-bar listing and program matching are permitted.

3. See Saved Program -- shows any pattern-generating programs that have been saved on the diskette. If no programs have been saved, the message "Sorry. There are no saved programs." appears. Note that regardless of how the programs were originally saved (as shapes or colors), programs displayed when the color monitor was selected at the beginning screen will use color statements and programs displayed using the black-and-white monitor option will use shape statements.

4. Save Program -- permits the saving of up to nine pattern-generating programs on the diskette. If there are already 9 programs saved, one will have to be deleted to make room for saving another program. A program is saved by entering a program in the same way that player-written programs are entered (see the section on "How to Enter Programs") except that it is done in the Teacher Option. When the program is completed, the computer will ask, if you want to save this program. If yes, the program is stored in the first empty space among the nine spaces available for saved programs.

5. Delete Saved Program -- allows a saved program to be deleted (erased) from the file on the diskette by typing the number of the program when asked. (If you are not sure of the program number, use option 3 to view the programs before using the delete option.)

6. (De)Select Saved Program -- allows the "select" to be turned on or off. When select is on, the player cannot write a hidden program or choose a difficulty level, but is given a specific program selected from those saved on the diskette by the teacher. (If there are no programs on the diskette, then select cannot be turned on.) This gives a teacher the option of having players work on the same pattern. To turn on the select option, the teacher types in the number of the program to be selected. Thus a teacher can save a program and select it and have each student in the class try to figure out the "pattern of the day." Option 3 will allow the saved programs to be examined first. Typing a zero "0" turns off the select option.

7. Turn Sound On/Off -- allows the sound to be turned on or off. The sounds at the opening screen, after successful and unsuccessful attempts to match the hidden pattern, and when input errors occur cannot be turned off. When the SPECTRUM program is first run, the sound is on.

8. Return to Main Menu -- brings the teacher to the main menu of the program.

Built in Instructions

The main menu of the game offers the option of Instructions. The diskette must be in the disk drive for running the instructions. Depending on whether or not a color monitor is being used, the instructions will use the words "colors" or "shapes." The instructions will not mention the bar-by-bar listing if the teacher has restricted it. The instructions are very important in order to gain a full understanding of how SPECTRUM works. Because of the detail and length of the instructions they are designed with a menu to enable the student to study various sections without going through all of the instructions each time.

POINTS OF STRATEGY

The best strategy will vary depending on whether other players or the computer have written the program that generates the "hidden pattern." The following notes on strategy will be helpful in general on any level. The examples below were constructed while working on level 8.

1. Always assume the shortest program has generated the pattern. Just as the scientist looking for patterns in the natural world proceeds by looking for the simplest, most conservative explanation consistent with existing evidence, SPECTRUM players should look for the shortest program that could have generated the colors (shapes) that are visible.

The computer writes programs using a random procedure that makes a short program more likely than a longer program. For example, say that you have requested BARS 1, 2, 4, 7, 11, 16 and 20 and found them all to be GREEN. Now, there are a large number of patterns that are consistent with this information:

The pattern might consist entirely of GREEN BARS - a two line program:

```
1 GREEN
2 F1*19
```

or it might consist of 19 GREEN BARS and a white colored fifth bar - a five line program:

```
1 GREEN
2 F1*3
3 WHITE
4 GREEN
5 F4*14
```

or it might be any number of other possible patterns. (This in fact is one of the crucial points about induction as a method of logical reasoning; it never decisively proves anything.) But although there are many possible programs, they are not all equally likely. The odds of the computer's random procedure generating the two-statement, all-green pattern are greater than for the pattern with the white fifth bar.

Even against a human opponent, whose programs are not random but are written so as to confuse and confound you, the strategy of assuming the shortest program makes good sense. If you assume a short program and you are wrong, you receive the counterbalancing benefit that since the hidden program is longer you will get more points. On the other hand, assuming a longer program -- and there are many more long programs than short ones -- risks the double loss of an unsuccessful try and the fewer points resulting from a short hidden program.

2. When playing against human opponents, be sure to use an unpredictable sequence for your initial bar requests. After all, your opponents may be devious, but they cannot read minds and predict in which order you will request to see the bars.

3. Use attempts to match the hidden program as tries for acquiring information as well as for matching the hidden pattern. An unsuccessful attempt costs as much in terms of the score as requesting to see a bar. (An unsuccessful attempt is subtracted from your score, while requesting to see a bar reduces the potential NUMBER RIGHT that you can get.) The attempt at a match can sometimes provide more information than a clue can and, of course, has the possibility of matching the pattern.

Examples of the 20-bar version of the game:

1. You have requested to see bars 1, 2, 7, 9, 11, 12, 18 and 20. The first four use the BOX and the second four use the ARROW. It looks like the program generating this patterns consists of the BOX, then 9 or 10 repetitions, the ARROW, and then 10 or 9 repetitions. The ambiguity seems to be in bar 10. If you ask to see bar 10 and then match the rest of the pattern, your score will be 75 (11 bars matched + 4 program statements = 15/20=75%). If, however, you simply try to match the pattern -- guessing the shape of BAR 10 -- half the time you will guess correctly and your score will be 80 (12 BARS right + 4 statements = 16; 16/20=80%). The other half the time you will guess incorrectly but match the pattern on your next try (assuming that BAR 10 is either BOX or ARROW) for a score of 75 (12 + 4 - 1 unsuccessful try = 15; 15/20=75%). So trying to match the pattern gives you a better score than requesting to see a bar half the time and gives the same score the other half of the time.

2. Your requests have revealed that bars 1, 2, and 7 are GREEN and 16, 19 and 20 are VIOLET. Again it makes sense to assume the pattern is a simple one: a series of GREENS bars and then a series of VIOLET bars. The question is -- where do the GREEN bars end and the VIOLET bars begin? Rather than taking stabs at it with bar requests, simply try all the bars up to and including 15 as GREEN. From the NUMBER RIGHT and NUMBER WRONG information you should be able to figure out the pattern so that you can get it on your next try (provided your assumption is correct that the pattern is some number of green bars and then violet bars). For example, if there were three wrong then you know that three of the greens should be violets and that therefore bars 1-12 would be GREEN and 13-20 VIOLET. If, instead, there were eight wrong, then 1-7 would be GREEN and 8-20 would be VIOLET.

3. Your bar requests have shown that bars 1, 2, 11, and 19 are all WHITE. You try a pattern of 20 WHITE bars and receive the information that the number wrong is one. Now, if you attempted to guess where the one non-white bar is, you would need - on the average - eight guesses and you would get - on the average - 8 bars right each time. On the other hand, you can converge on the non-white bar using the list method of matching by the following procedure: fill in

the remaining unrevealed bars less than 11 with white and the ones above 11 with some other color, say violet. If there are seven right, then you know that the non-white bar must fall between 1 and 10. (This follows from the fact that you know there is only one non-white bar and that, therefore, all the white bars you just entered except one must be correct.) If there are eight right, these must be all the whites and the non-white will also be non-violet and be located in bars 11-20. And if there are nine right, then it means that there is a VIOLET somewhere in bars 11-20. Next, you try entering all whites except for four bars of another color (not VIOLET, unless the previous try confirmed that VIOLET was the non-white color), and then you try with two bars of another color, and finally with one of another color. At most, this procedure will take you five tries to converge on the pattern, which results in a higher score than would be obtained by using eight bar requests.

SUGGESTIONS FOR CLASSROOM USE

Before introducing students to SPECTRUM, you may want to discuss with them what patterns are, where they can be found, how they are used, and why being able to recognize patterns is important.

Begin by explaining that patterns are everywhere -- in clothing, in the trees, in a book, and in the sky. Ask students to name where patterns can be found.

Generate a discussion about how patterns are used and why they are important. Here are a few ideas to help you get started:

1. Patterns may be used to create things. For example, a recipe is a pattern. And, more important, if the pattern (recipe) is followed very carefully, the result will be the same -- every time.

2. Patterns are used as models -- a builder needs a blueprint to create a building; a musician writes and reads a musical score to create a song or a symphony.

3. Patterns help predict things. Financial analysts study the pattern of the stock market to plan their investments; weather forecasters study the cloud, temperature and humidity patterns to predict tomorrow's weather; history is studied because historians have found that history does repeat itself; some people study voter samples to predict the outcome of elections; and doctors try to discover the pattern of our symptoms so they can prescribe the best remedy possible.

4. Patterns are predominant in the area of mathematics -- natural numbers, odd numbers, even numbers, perfect squares and many more. You may want to introduce your students to some of the more popular math games described in Patterns -- What Are They? by William Shimek (Lerner Publications Company, Minneapolis, Minnesota, 1969.)

SPECTRUM lets students practice these problem-solving strategies:

1. Information gathering.
2. Looking for a pattern or sequence.
3. Analyzing.
4. Scanning for clues.
5. Estimating, predicting, and projecting.
6. Working backwards.
7. Examining assumptions.
8. Identifying multiple solutions.

In addition, when students play SPECTRUM on their own they learn the important skill of thinking on their own.

About the Worksheets

There are three master worksheets provided which may be copied and distributed to your students. The first one, the STUDENT RECORD WORKSHEET may be used by a student or group of students to record the progress of each round.

Two game plan worksheets, the 20 bar version and the 40 bar version are also included. These worksheets will be very handy for students keep track of the clues as well as recording information they discover as they attempt to match the pattern. Each student may need a few copies of the worksheet since it may take many tries to figure out a given pattern. Scratch paper will also be needed as they take notes from one step to the next.

SPECTRUM: Patterns and Programs
Student Record Worksheet

Name (s)

Player 1 _____ Player 3 _____

Player 2 _____ Player 4 _____

Class _____ Date _____

Circle Options Which Applied To Your Game:

Monitor: Color Black & White

Computer: Apple TI

Create a Pattern

Number of Bars

20 40

Number of Colors/Shapes

3 4 5 6

Play a Game

Difficulty Level

1 2 3 4 5 6 7 8 9

Special Level _____

Final Score:

Player 1: Score _____

Number of attempts _____

Number of statements in hidden program _____

Player 2: Score _____

Number of attempts _____

Number of statements in hidden program _____

Player 3: Score _____

Number of attempts _____

Number of statements in hidden program _____

Player 4: Score _____

Number of attempts _____

Number of statements in hidden program _____

SUNBURST COMMUNICATIONS © 1984

SPECTRUM: Patterns and Programs
GAME PLAN WORKSHEET - 20 Bar Game

Name _____ Date _____

Bar Number	Turn # clue	attempt	Turn # clue	attempt	Turn # clue	attempt	Turn # clue	attempt
1								
2								
3								
4								
5								
6								
7								
8								
9								
10								
11								
12								
13								
14								
15								
16								
17								
18								
19								
20								
	Right _____		Right _____		Right _____		Right _____	
	Wrong _____		Wrong _____		Wrong _____		Wrong _____	

SUNBURST COMMUNICATIONS © 1984

SPECTRUM: PATTERNS AND PROGRAMS
GAME PLAN WORKSHEET - 40 Bar Game

Name _____ Date _____

	Turn #			Turn #			Turn #			Turn #	
	Clue	Attempt		Clue	Attempt		Clue	Attempt		Clue	Attempt
1			21			1			21		
2			22			2			22		
3			23			3			23		
4			24			4			24		
5			25			5			25		
6			26			6			26		
7			27			7			27		
8			28			8			28		
9			29			9			29		
10			30			10			30		
11			31			11			31		
12			32			12			32		
13			33			13			33		
14			34			14			34		
15			35			15			35		
16			36			16			36		
17			37			17			37		
18			38			18			38		
19			39			19			39		
20			40			20			40		

Number Right _____ Number Right _____

Number Wrong _____ Number Wrong _____

SUNBURST COMMUNICATIONS © 1984

APPLE: WORKING WITH THE COMPUTER

1. Turn on the television or monitor.

2. Insert the diskette into the disk drive with the label facing up and the oval shape inserted first.

3. Close the door to the disk drive.

4. Turn on the Apple II. (The on-off switch is on the back left side of the computer.)

5. You will see a red light on the disk drive turn on. If the disk drive light does not turn off in about 10 seconds, turn the Apple off and make sure your diskette is placed correctly in the disk drive.

6. SUNBURST will appear on the screen followed by the program name.

7. Follow directions given in the program.

8. Holding down the CONTROL (CTRL) key and pressing "E" will end what you are doing and start over.

Shutting Off the System

1. Remove the diskette from the disk drive and return it to its place of storage.

2. Turn off the Apple.

3. Turn off the television or monitor.

TI99/4A: WORKING WITH THE COMPUTER

1. Turn on the television or monitor.

2. Turn on the computer.

3. Choose the tape you wish to use.

4. Insert the tape in the tape recorder (the label of the program you want should be facing up).

5. Press any key to begin.

6. Press the "1" key for TI BASIC.

7. Type OLD CS1 and press the ENTER key (NOTE: 1 is the number one).

8. The following will appear on the screen:

   *Rewind cassette tape then press ENTER.

   (When you see this, rewind the tape, set the counter to "0" (zero), advance the tape to 9 and then press ENTER.)

9. Follow the rest of the directions on the screen.

10. When the " > ■ " appears on the screen type RUN and press ENTER.

"WHAT HAPPENS IF...?" -- SUNBURST COURSEWARE AND WARRANTY

1. What happens if a program will not load or run?
   Call us on our toll-free number and we will send you a new tape or diskette.

2. What if I find an error in the program?
   We have thoroughly tested the programs that SUNBURST carries so we hope this does not happen. But if you find an error, please note what you did before the error occurred. Also, if a message appears on the screen, please write the message down. Then fill out the evaluation form or call us with the information. We will correct the error and send you a new tape or diskette.

3. What happens if the courseware is accidentally destroyed?
   SUNBURST has a lifetime guarantee on its courseware. Send us the product that was damaged and we will send you a new one.

4. How do I stop a program in the middle to go on to something new?
   The Apple version can be ended at any time by holding the CONTROL button and pressing the E key. To change diskettes, select the End option on the menu and insert a new diskette. On the TI, type FCTN 4 and the program will be interrupted. Then, a new tape may be loaded or SPECTRUM may be started over again by typing RUN.

5. Can I copy this program?
   The material on the diskette or cassette is copyrighted. You should not copy the courseware.

6. Can I load this program into the machine and remove the diskette/ cassette in order to run it on more than one machine?

   On the Apple, the program needs to access the diskette during the run and, therefore, the diskette must remain in the disk drive while using the program.

   On the TI, the program is completely loaded into the computer when the " > ■ " appears on the screen. At this time you may remove the cassette tape and load it into another machine.