Music Input Program Plays 3-Tone Music in Basic

Tutorial By George Moses
Program by Brett Bilbrey, modified by George Moses and Bob Weber

1.1234567890123456789012345678901234567890123456789012345678901234567890123456789012345678901234567

Line 1 is what is called a "REM" statement. That is, a line number followed by a period, then some information. When a period follows a line number, anything after the period is completely ignored by the computer. However, anything following the period takes up memory space, and if we poke data into these memory locations and remember which locations the data occupies, the computer can go back and read the data later through the peek function and perform the program instructions using this information to play music, do graphics or anything you tell it to do. Storing data in the text area of the Bally (memory locations —24576 thru —22777) isn't a new idea. It was brought to light thanks to Dave Ibach's excellent tutorial on the subject, page 78 Arcadian 1979. ***You can put up to 14 REM statements in this program with exactly 97 numbers after the period. Number them 1 thru 14 and don't forget the period. The rest of the program follows. REMEMBER the character "c"means divided by, and the character "b" means times as the printer doesn't have the × and + signs.***

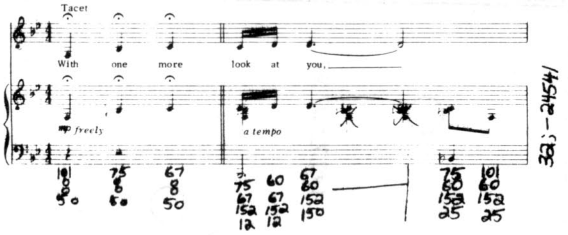
```
  50 CLEAR ;E=-24573;Z=0
  60 PRINT #1,Z,":",E+Z,"=",;INPUT " "J;IF J>256GOTO 100
  65 J=J-127;IF J<0J=J-1
  70 %(E+Z)=J
  80 Z=Z+1;IF Z>95Z=0;E=E+101
  90 GOTO 60
 100 INPUT T;IF T=0NT=3;GOTO 60
 105 NT=0;&(16)=49;&(22)=136;&(21)=15;FOR A=-24574TO -23160STEP 101
 110 FOR C=ATO A+92STEP 4;&(17)=%(C)c256+127;&(18)=%(C+1)c256+127;&(19)=%(C+2)c2
56+127
 120 FOR D=1TO (%(C+3)c256+127)bT;NEXT D
 130 IF C>E+Z-8GOTO 150
 140 NEXT C;NEXT A
 150 &(21)=0;&(22)=0;&(16)=0;&(17)=0;&(18)=0;&(19)=0
 160 NT=3;GOTO 100
```

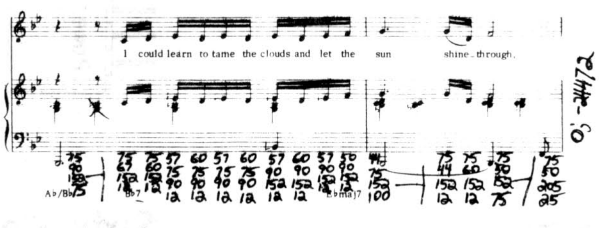
Now get yourself some sheet music and start programming music. See the sheet I've included for an example. The columns of four numbers under each note played each constitute a chord of three notes (the top three numbers) plus a duration input (the bottom number). The duration input allows you to play any length of note required by the sheet music. See the note and duration input chart at right.

Now, run the program and you'll see the numbers 0; —24573 at the top left of your screen. That's the memory location of your first note input to the A voice of your sound synthesizer. Put the number of the note you want in voice A and hit GO. Then the next memory location will come onto the screen and wait for you to input the note value for the B voice, and next you'll input the C voice. Then, the fourth input will be your duration value for that chord of notes.

So, there are four inputs for each chord. If a chord has only one note (or no notes as in a rest) just input zero into the voices you wish silent. IMPORTANT! If you input the same note into the same voice in two consecutive chords the result will be one continuous note carried for two beats, or as many beats as you wish to carry it out. Often this is desirable, but if you want the same note played two distinct times consecutively, then you must switch voices, playing the next note in the other register and set the first register to zero or use it to play another note in the chord. You'll see some examples of this on the sheet music I included. In the second measure you'll see the note C (67) carried out for two 16th notes in a row in the same register making it a continuous note, while I swapped registers with the B (60) to play two distinct notes consecutively.

Now, go ahead and input the numbers in the sheet music going top to bottom on the columns under each note. When you get to the end of the first bar your last memory location number on the screen should be as noted in the right margin of the sheet music 32; —24541. Now, INPUT any number larger than 256 and the computer will automatically GOTO 100 and prepare to play the music. Input T. 1 will make it play the fastest. A larger T will make it play slower. The music will play up to the end of your data and stop. This way you can proofread what you've input up to that point. The program will ask you to input T again. If you want to hear the music again input T and hit GO. But, if you want to resume inputting data input zero, hit GO and you'll automatically GOTO line 60 to take up where you left off.

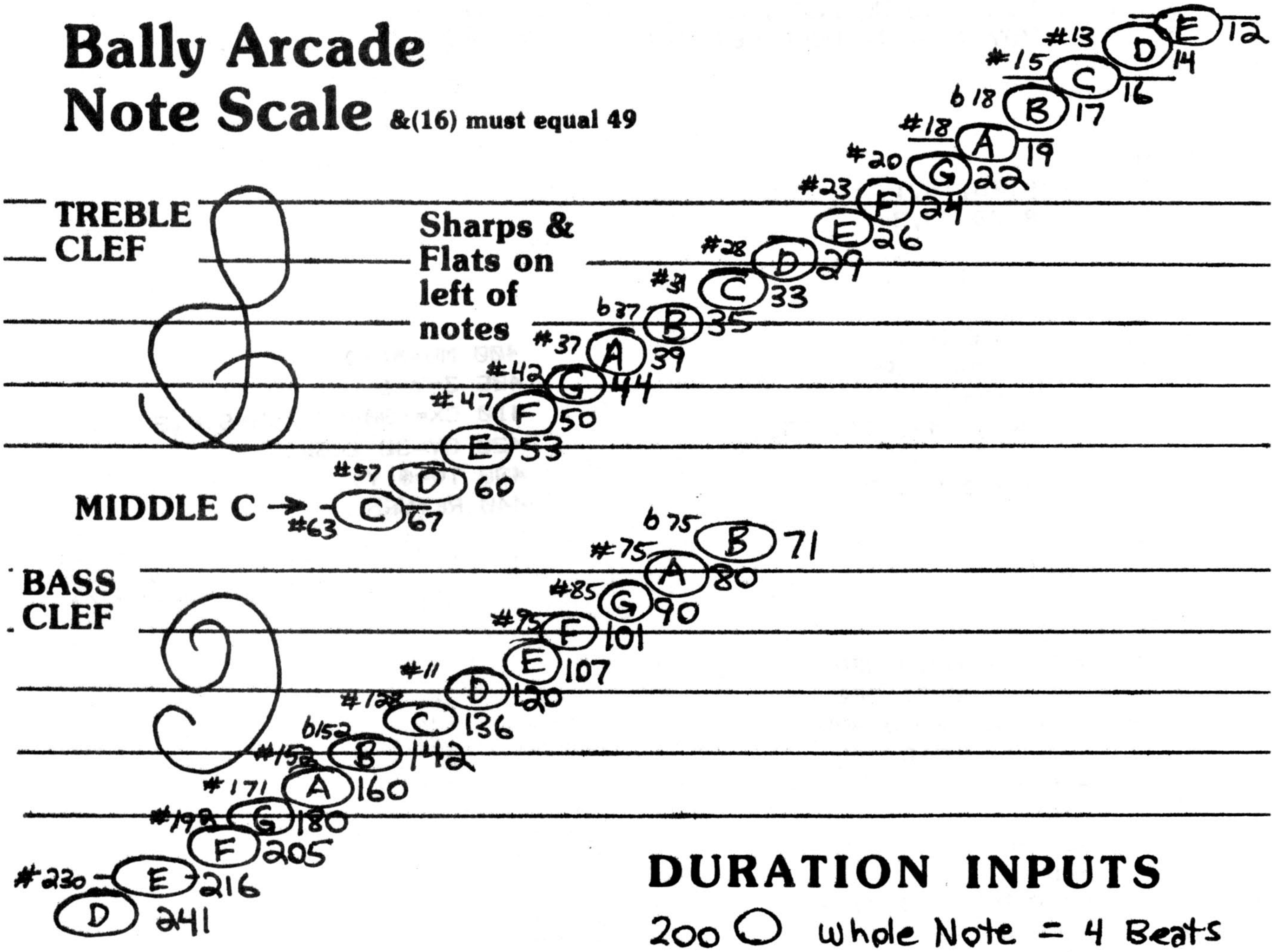



I could learn to tame the clouds and let the sun shine through.
Ab/Bb
Bb7
Ebmaj7
0; -24472

Moses, George. "Music Input Program Plays 3-Tone Music in Basic," ARCADIAN 2, no. 7 (May 19, 1980): 62-63.

Bally Arcade Note Scale

&(16) must equal 49



DURATION INPUTS

200	𝅝	Whole Note = 4 Beats
100	𝅗𝅥	Half Note = 2 Beats
50	♩	Quarter Note = 1 Beat
25	♫	Eighth Note = ½ Beat
12	♬	Sixteenth Note = ¼ Beat
6	𝅘𝅥𝅰	Thirty Second = ⅛ Beat

What about correcting mistakes? Simple. Just input the memory location of the beginning of that bar and input the data again. Say the first bar was wrong. Just HALT the program and type in the beginning memory data like this: Z = 0; E = —24573—Z;GOTO 60 and this will get you back to the beginning of the first bar. For this purpose it's advisable to put the last memory readout at the right margin of each bar as per the example included. The numbers at the end of the first bar would get you set up at the beginning of bar 2 if you were to have a glitch there and needed to input the correct data. It is IMPORTANT to proof-listen to each bar before going on because you cannot input corrections in the middle of a song. You can start in the middle, but you have to keep inputting data till the end. The Bally looks at two bytes of memory at a time when inputting and while you correct one note, you're creating a glitch in one of the following notes, so you have to keep rolling to cover up the whole performance perfectly.

After you input your last note you should alter line 130 to read the ending location to stop the music at the end of data. If you don't you'll get some awful sounds when the computer runs out of notes to play. Just halt the program and ask the computer to PRINT E + Z-8. The answer will be a nvgative number. Take this number and put it in line 130 in place of the E and Z-8 variables and you have a complete program.

Another line you may want to change for convenience is line 100. Once you decide the best value of T for that song, just set T equal to that number permanently in the line. Also you may want to do the following with line 160 FOR M = 1 TO 2000; NEXT M; GOTO 100. This will give you a five second delay at the end of the song and automatically begin the music again.

To save on tape with automatic RUN just get your recorder ready, then type in *NT = 1;CLEAR;:PRINT;LIST; PRINT":RETURN;CLEAR;NT = 0;GOTO 100* Turn on your tape recorder, then hit GO and watch it list. The listing of the REM statements will be mostly question marks, but don't worry. The computer understands what they are through the peek command in the program. Any questions or new discoveries about this music program will be welcomed by all of us of the Bally Users' Group of Michigan at 110 E. North St., Brighton, Michigan 48116.

Moses, George. "Music Input Program Plays 3-Tone Music in Basic,"
ARCADIAN 2, no. 7 (May 19, 1980): 62-63.