

*** PROGRAMMING LOW-RES GRAPHICS IN Applesoft BASIC ***

*** Getting Started ***

Boot into your favorite version of DOS or ProDOS that supports Applesoft BASIC low-res graphics mode (GR command). Most copies of DOS, ProDOS or the Apple System Utilities Disk include Applesoft BASIC.

You can also enter Applesoft BASIC by pressing control-reset during the boot sequence. However, you will not be able to save programs using this method. Enjoy your ephemeral art... if you want to get even more existential, simply type the commands at the prompt without writing them as a full program with line numbers...

To save your programs you will need a disk formatted for your version of DOS/ProDOS, or you may use a disk with other programs already on it that has some space left. If you are using an emulator, you can use a blank disk image, or use the one provided.

*** Writing Your First Program in Low Res Graphics Mode ***

The "GR" command is used to enter low-res graphics mode. In this example, let's start a new program in graphics mode. Use the "HOME" command to clear any leftover text on the screen before drawing, it will make your image look a little nicer. (Optional)

```
1 HOME
```

```
2 GR
```

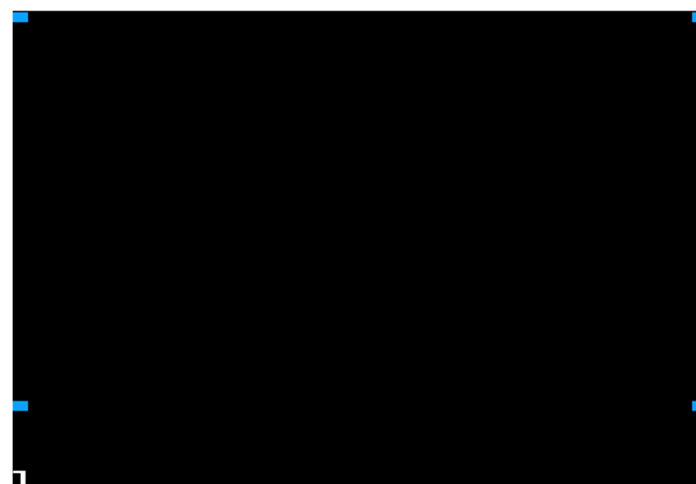
Now we can start drawing. We will need to set our first color. The colors are numbered 0-15. See the color chart provided for reference. In this example, let's start with Color # 6 aka 'medium blue'

```
1 30 COLOR = 6
```

You can draw points and lines in low-res mode. The screen consists of 40 X 40 rectangular pixels, numbered 0 through 39.

Let's draw a point to show each corner. Points are plotted using x,y coordinates, with 0,0 being the upper left-most point. We can add a comment or remark (REM) to remind ourselves of what we are drawing for future reference.

```
1 40 REM DRAW CORNERS
1 50 PLOT 0,0
1 60 PLOT 0,39
1 70 PLOT 39,0
1 80 PLOT 39,39
```



What point(s) would you plot to mark the center?

```
1
1
```

If you are using a disk or disk image, this would be a good time to save your work. Use the SAVE command followed by the name for your program.

For example:

```
] SAVE DEMO
```

To see what you have typed so far use the "LIST" command. To run the program use "RUN."

If you typed LIST and noticed that it was hard to read your program code in the 4 lines of text at the bottom under the image space, you can use the "TEXT" command to switch back to text mode while you are writing code. If you want to clear the screen first, type "HOME."

For example:

```
] TEXT
```

```
] HOME
```

```
] LIST
```

Now, let's draw some lines. Use the "HLIN" command to draw horizontal lines and "VLIN" to draw vertical lines. Let's change the color to orange and connect the dots.

```
] 85 REM DRAW LINES
```

```
] 90 COLOR = 9
```

```
] 100 HLIN 0,39 at 0
```

The HLIN command above will draw a line from 0,0 to 0,39. You could plot each point individually, but drawing lines can save you lots of time. VLIN works the same way except it draws vertical lines.

```
] 110 VLIN 0,39 at 0
```

This will draw a vertical line at the leftmost part of the screen. Let's run the program again and see what it looks like:

```
] SAVE DEMO
```

```
] RUN
```



Oh no! Where did the corner dots go??!!

Points or lines are drawn in the order you programmed them, so anything you draw on the same spot will replace what was previously there. If you wanted the dots to stay on the screen, you would need to draw them after the lines.

For now, let's just finish drawing the box around the edges of the screen.

```
] 120 HLIN 0,39 at 39
```

```
] 130 VLIN 0,39 at 39
```

You can then SAVE and RUN your program to see the whole box.

Now you have the commands to draw dots and lines and change colors. Use the graph paper provided to design your own image, or try writing a program to recreate one of the sample images provided.

*** Tips and Tricks ***

** Filling the Screen **

If you want a colored background, you can use a function to fill the screen with a background color before drawing your image. Here is an example:

```
5 REM SCREEN FILL
10 TEXT
20 GR
30 COLOR = 6
40 FOR N = 0 TO 39
50 HLIN 0,39 AT N
60 NEXT
```

**** Animation ****

You can mimic the effect of animation using a few tricks.

*** Drawing order ***

If you want your image to be drawn in a particular order, for example, the background first, then a circle, then a smiley face on the circle, write your program in that order.

It takes longer to draw images pixel by pixel rather than using lines, but if you do that, and plan carefully, you can write your image in order to have it appear more like it is dissolving in. You could plot all pixels of one color, or plot the image from top to bottom to look like it is being drawn line by line, etc.

**** Simple Color Changes ****

You can use a random number generator and a function to make your images flash between different colors. See the Christmas tree program provided for an example of this (TREE3).

Using "GR" will clear the screen and can give a "flipbook" type effect.

**** Faux-nimation ****

For more sophisticated looking animation, you can "move" your pixels across the screen. For this you will need to plan ahead a bit and make some of your points variables, then move them +1 pixel across the screen. Look at the xyz program for an example. Use a pause function to add time to speed up or slow down your images. See the "ANIM" program for an example of using this to draw rain from the clouds.

**** Using a Timer ****

You can use a FOR loop as a counter, which will pause before running the next part of your program, giving the illusion of animation. See the "LINE" program for an example of this.

What other ways can you use these in your programs, or what other ideas do you have?