

# Creating HP 38G Aplets

This article explores a simple aplet and shows how to construct an aplet called PolySides.

by James A. Donnelly

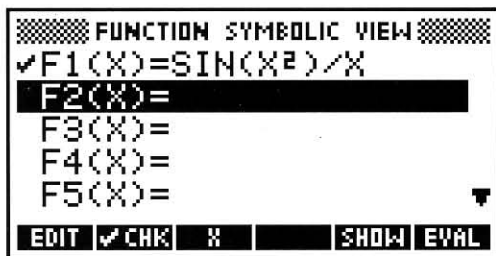
In designing an aplet for the HP 38G calculator, it's important to remember two guiding principles of the HP 38G design. First, the design recognizes that in a classroom setting there is (and should be) a large discrepancy between the amount of information entered into a graphing calculator and the amount of information produced by the calculator. We sought to minimize the calculator input required of the student and the teacher and maximize the returned information. Secondly, we sought to exclude irrelevant possibilities. By reducing irrelevant choices you focus the user's attention on the subject of interest and avoid distractions from things that don't matter. Many choices made during the design of the HP 38G were based on this goal.

Aplets contain information and have views. The information in an aplet consists of every major piece required to produce the views: the equations, setup information, mode information, sketch or text annotations, and attached libraries or programs. In this article we'll explore a simple aplet, then look at an aplet called PolySides and examine how it was constructed.

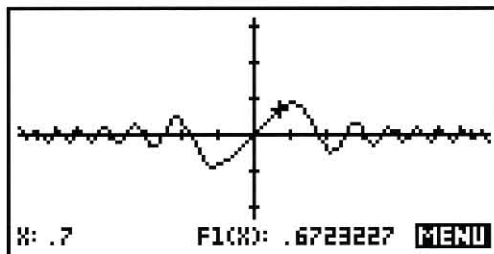
## Using Built-in Aplets

When the HP 38G is first turned on, the built-in aplets are empty. There's no equation, note, or sketch. When the user adds this information, these aplets come alive. You can save an aplet at any time, so it's easy to start one project, change in midstream to another, then come back to the first. (Because the appearance of the screens depends on the information you enter, the screens on your calculator may look different from the example screens in this article.)

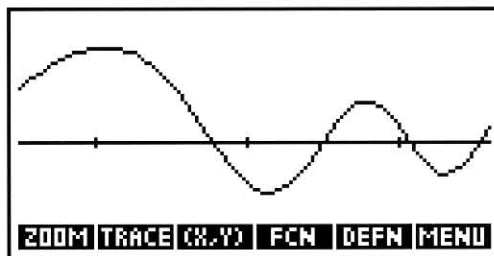
A simple way to illustrate the aplet concept is to explore the equation  $\sin(x^2)/x$ . Select the function aplet, press **SYMB**, and enter the equation. ➤



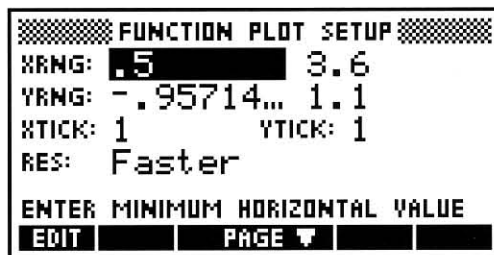
Now press **PLOT** to see the plot using the default plot parameters. ➤



Use the box option under **ZOOM** to look at a smaller area of the plot. ➤



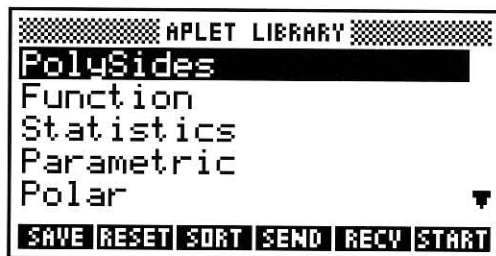
You can see the plot scale by pressing [shift]PLOT to see the plot setup view. ➤



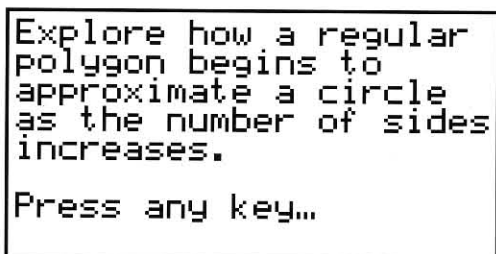
At this point, you have an aplet that's completely dedicated to your interest in the function  $\sin(x^2)/x$ , and the aplet can be saved under a unique name or transmitted to another HP 38G or a computer. When the aplet is restarted on the original or another HP 38G, all modes and scales are set just the way they were saved.

## Exploring the PolySides Aplet

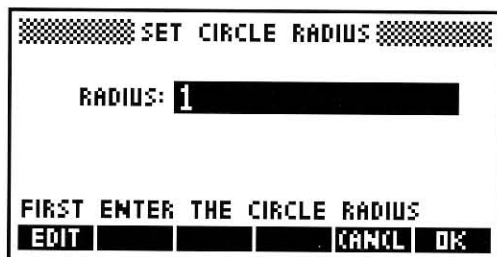
The PolySides applet is designed to explore how a regular polygon can approximate a circle as the number of sides increases. PolySides can be loaded from a disk or another HP 38G in a single operation from the LIB catalog. Once loaded, PolySides appears in the applet library. ▼



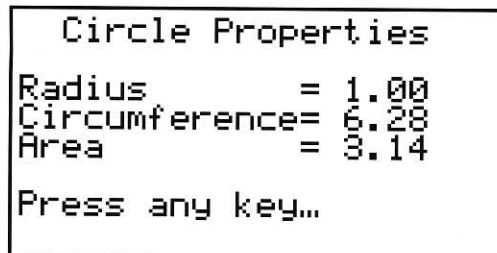
We are now just a single keystroke away from exploring the applet. To begin the exploration, press START. The motivation for the lesson is displayed first. ▼



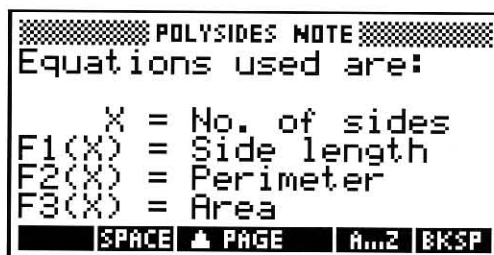
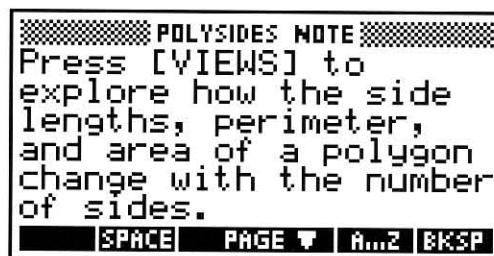
After the introduction has been read and a key pressed, the next step is to enter the radius of the circle to be approximated. ▼



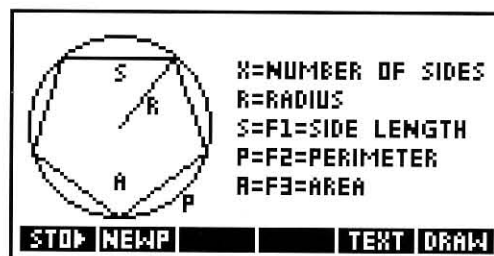
After the radius has been entered, the properties of the circle are displayed. ▼



After the circle properties have been viewed, the note view of the applet is displayed. The PAGE menu key switches between the pages of the note. ▼



To see a sketch of the problem, press SKETCH. ▼

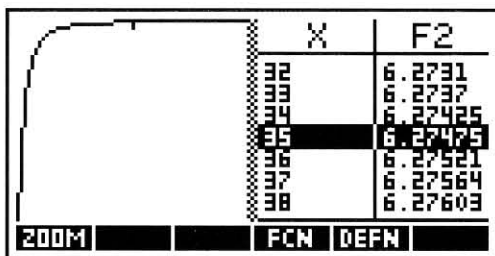


So far we have seen a pretty complete summary of the lesson with less than a dozen keystrokes. Now we can begin to explore how many sides are needed to approximate a circle.

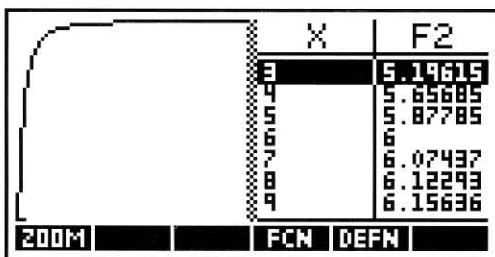
Our central point of departure for PolySides is VIEWS (as mentioned in the note view). ▼



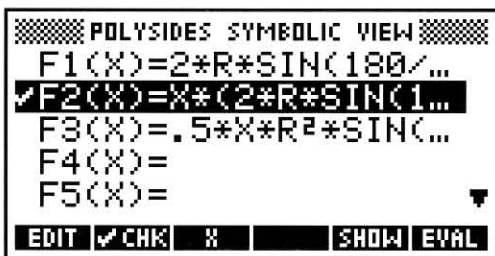
These choices let you determine what aspect of a polygon approximation to a circle you'd like to explore. To explore how the perimeter of the polygon approximates the circle, move the highlight down to Perimeter and press OK. The plot-table view is presented automatically. ▼



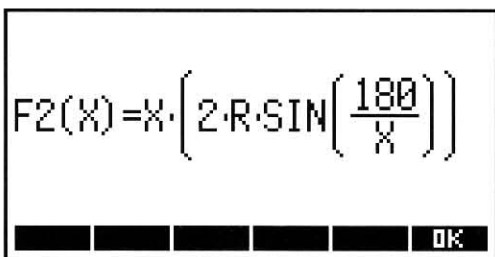
This is one of the HP 38G's split views, showing the plot and numeric views of the perimeter approximation at the same time. The variable X represents the number of sides, and the equation stored in F2 returns the perimeter as a function of X. Pressing the left or right arrow key moves the cursors for the plot and numeric views simultaneously. When the cursor is at the left edge of the plot, the table on the right shows the rapid change of the perimeter from a triangle to a nonagon. ➤



To see the equation, just press SYMB. ➤

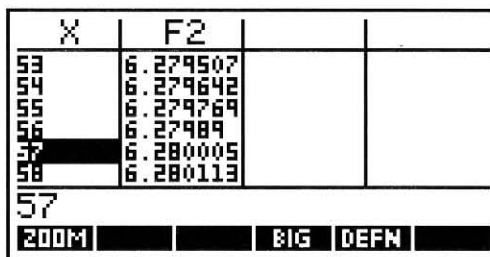


The symbolic view is where apilet equations are stored. The check mark beside F2 indicates that when a plot or numeric view is displayed, F2 will be used. Another view of the equation is available by pressing SHOW. ➤



Notice that after the introduction there is no fixed order of events. You can change to any of the views at any time or press HOME at any time to do a short calculation. Part of the design philosophy of apilets is to let the student explore the lesson at will, without following any particular algorithm. Once you've explored the approximations to the perimeter and area, and perhaps explored the effect on side lengths,

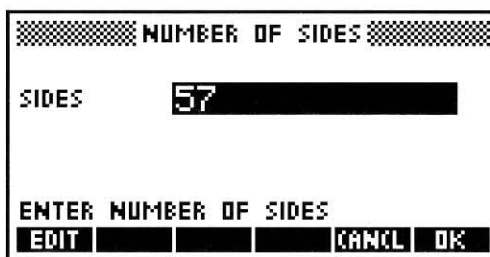
you may decide that a polygon with 57 sides yields a fair approximation to the circumference of a unit circle. ➤



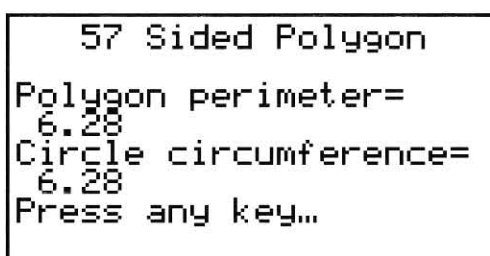
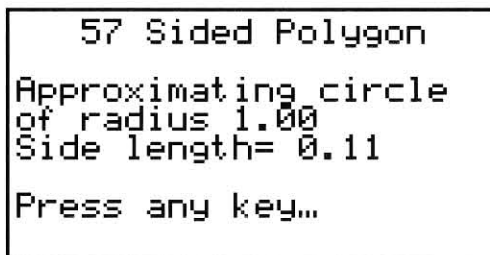
Now you can look at a summary of a 57-sided polygon by selecting Polygon Props (for polygon properties) from the VIEWS choices. ➤

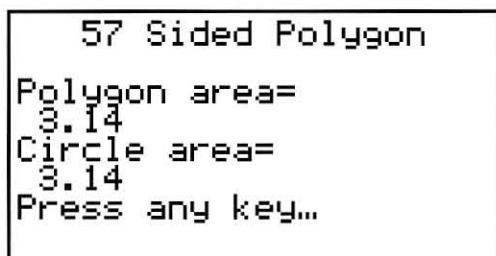


Press OK to select this choice. Since the cursor was last on X = 57, the number of sides defaults to 57. ➤



After the user accepts (or alters) the number of sides, the polygon data is displayed. ➤





Now we've observed that a 57-sided polygon does a fair job of approximating a unit circle. More explorations are possible. For instance, if the circle has a very large radius, how many more sides are required for a polygon to closely approximate the circle's area?

After the last polygon property screen has been acknowledged with a keystroke, the **VIEWS** menu is displayed. ➤



This lets you start with another circle or go back to find a different approximation.

Notice that the keystrokes involved for the entire exploration have been oriented to the views. We haven't entered a single bit of setup or scaling information beyond the radius of the circle being approximated. This is the goal of a well-formed applet; more attention is directed to the lesson and less attention is directed to the mechanics of the calculator. You can imagine a teacher distributing this applet in a classroom for part of a day's lesson.

If you're familiar with the HP 48G/GX calculators, you'll notice that variables like EQ and PPAR have never been mentioned, even though we changed the equation and plot scaling parameters several times. Furthermore, we never went near an input form to set the scale manually (although we could have done so by pressing [shift] PLOT to display the plot setup view).

### Building the PolySides Applet

PolySides was constructed using the function applet and five user programs attached with the SETVIEWS command.

Beginning with a reset function applet, the equations, note, and sketch were entered directly on the calculator. The equations are:

- Side Length      $F1(X) = 2 \cdot R \cdot \sin(180/X)$
- Perimeter        $F2(X) = X \cdot 2 \cdot R \cdot \sin(180/X)$
- Area              $F3(X) = .5 \cdot X \cdot R^2 \cdot \sin(360/X)$

Then five programs were entered, one for each entry in **VIEWS**. After the programs were entered, they were attached to the applet with the SETVIEWS command. For convenience during applet development, a separate program, SetPolyViews,

was entered that contains the SETVIEWS command. SETVIEWS allows you to customize the **VIEWS** key with your own entries and selected entries from the default **VIEWS** choices. SETVIEWS takes triplets of arguments. Each triplet contains the prompt that will appear in the **VIEWS** list, a program name, and a number corresponding to the view that should be displayed after the program is executed. The SETVIEWS command for PolySides looks like this:

```
SETVIEWS
"Start";".Poly1";8;
"Side Lengths";".Poly2";16;
"Perimeter";".Poly3";16;
"Area";".Poly4";16;
"Polygon Props";".Poly5";7;
```

When Side Lengths is selected, the program .Poly2 is executed, then view number 16 is displayed. View number 7 is the **VIEWS** list, view number 8 is the notes view, and view number 16 is the split plot-table view. A SETVIEWS convention is that if a prompt is named Start or Reset, it is associated with the START or RESET menu key in the **LIB** catalog. The PolySides applet takes advantage of this feature so that pressing START in the **LIB** catalog gets a student underway in a single keystroke.

The programs are listed below. (Note that the translation codes used are the same as translate code 3 on the HP 48G/GX.)

#### .Poly1 (Start)

ERASE:	<i>Clear the display</i>
DISP 1;"Explore how a regular":	<i>Display the</i>
DISP 2;"polygon begins to":	<i>motivation for</i>
DISP 3;"approximate a circle":	<i>the applet</i>
DISP 4;"as the number of sides":	
DISP 5;"increases.":	
DISP 7;"Press any key...":	
FREEZE:	
IF R\<=0 THEN 1\ >R END:	
INPUT R;"SET CIRCLE RADIUS";	<i>Prompt for the</i>
" RADIUS:";	<i>circle radius</i>
"FIRST ENTER THE CIRCLE RADIUS";R:	
ERASE:	<i>Clear the display</i>
2\ >Digits: Fixed\ >Format:	<i>Set FIX 2 display</i>
	<i>mode</i>
DISP 1;" Circle Properties":	<i>Display the screen</i>
	<i>title</i>
DISP 3;"Radius = " R:	<i>Display each</i>
	<i>property</i>
DISP 4;"Circumference = " 2*\pi*R:	
DISP 5;"Area = " \pi*R^2:	
DISP 7;"Press any key...":	
FREEZE:	<i>Wait for a key</i>
	<i>press</i>
Standard\ >Format:ERASE	<i>Restore standard</i>
	<i>display format,</i>
	<i>clear the display</i>

Programs .Poly2 through .Poly4 do similar jobs. Each validates the value for R (the radius), selects the appropriate equations in the symbolic view, calculates the plot scale parameters, and sets the initial values for the numeric view. Note that the plot scale is calculated to fit above the menu. The menu occupies 1/8 the display, so an adjustment of 1/8 the calculated vertical range is made to the vertical scale.

### **.Poly2 (Side Lengths)**

```
IF R\<=0 THEN 1\|>R END:      Ensure a reasonable value for R  
  
CHECK 1:                      Select equation F1  
UNCHECK 2:  
UNCHECK 3:  
3\|>Xmin:                     Set the plot scale  
67\|>Xmax:  
F1(Xmax)\|>Ymin:  
F1(Xmin)\|>Ymax:  
Ymin-.125*(Ymax-Ymin)\|>Ymin: Fit the plot above the menu  
  
3\|>NumStart:                 Set the numeric view to begin at 1 and increase in steps of 1  
  
1\|>NumStep:
```

### **.Poly3 (Perimeter)**

```
IF R\<=0 THEN 1\|>R END:      Select equation F2  
UNCHECK 1:  
CHECK 2:  
UNCHECK 3:  
3\|>Xmin:  
67\|>Xmax:  
F2(Xmin)\|>Ymin:  
F2(Xmax)\|>Ymax:  
Ymin-.125*(Ymax-Ymin)\|>Ymin:  
3\|>NumStart:  
1\|>NumStep:
```

### **.Poly4 (Area)**

```
IF R\<=0 THEN 1\|>R END:      Select equation F3  
UNCHECK 1:  
UNCHECK 2:  
CHECK 3:  
3\|>Xmin:  
67\|>Xmax:  
F3(Xmin)\|>Ymin:  
F3(Xmax)\|>Ymax:  
Ymin-.125*(Ymax-Ymin)\|>Ymin:  
3\|>NumStart:  
1\|>NumStep:
```

### **.Poly5 (Polygon Properties)**

```
INPUT X;"NUMBER OF SIDES";"SIDES";"ENTER NUMBER  
OF SIDES";X:                  Sides prompt  
ERASE:                         Clear the display  
Standard\|>Format:            Set standard format  
  
DISP 1;" " X " Sided Polygon": Display the screen title  
  
2\|>Digits: Fixed\|>Format:    Set FIX 2 format  
DISP 3;"Approximating circle": Display circle  
DISP 4;"of radius " R:        radius and poly-  
DISP 5;"Side length= " F1(X): gon side length  
DISP 7;"Press any key...":  
FREEZE:                        Wait for a key press  
  
ERASE:                         Clear the display  
Standard\|>Format:            Set standard format  
  
DISP 1;" " X " Sided Polygon": Display the screen title
```

```
Fixed\|>Format:                Set FIX 2 format  
DISP 3;"Polygon perimeter=":   Display the  
DISP 4;" " F2(X):             perimeter and  
DISP 5;"Circle circumference=": circumference  
DISP 6;" " s*\pi*R:  
DISP 7;"Press any key...":  
FREEZE:                        Wait for a key press  
  
ERASE:                         Clear the display  
Standard\|>Format:            Set standard format  
  
DISP 1;" " X " Sided Polygon": Display the screen title  
  
Fixed\|>Format:                Set FIX 2 format  
DISP 3;"Polygon area=":        Display the area  
DISP 4;" " F3(X):  
DISP 5;"Circle area=":  
DISP 6;" " \pi*R^2:  
DISP 7;"Press any key...":  
FREEZE:                        Wait for a key press  
  
Standard\|>Format: ERASE      Restore standard display format, clear the display
```

## **Building Your Own Aplets**

Your own aplets can be as simple as the first example, somewhat more involved (like the PolySides aplet), or very involved, with more complex user programs. Creating a new base aplet that can be downloaded into the HP 38G requires the use of System RPL programming beyond the built-in user programming language.

**Basic Steps.** The basic steps for building an aplet are:

- Pick an appropriate built-in base aplet—function, polar, parametric, etc.
- In each of the views, enter the information that applies. For instance, you would put your equations in the symbolic view, plot parameters in the plot setup view, and so on.
- Put a sketch of the problem in the sketch view and a description of the problem in the note view.
- For a fancier aplet, use SETVIEWS to attach user programs to the **VIEWS** key. Remember that it may be handy to have a separate program that stores the SETVIEWS command while you're developing the aplet.

**Flow of Control.** In general, the user of the aplet should determine the flow of control, so that a problem can be explored freely. The PolySides aplet provides a little extra guidance by using the program .Poly1 attached to the start prompt in **VIEWS**. .Poly1 displays an initial screen, prompts for the circle radius, displays the circle's properties, and finally exits to the note view as directed by the SETVIEWS command. In general, the user should be able to navigate freely among the choices in **VIEWS** or the various views, or go home and come back.

**Shared Variables.** The global variables A to Z, Z1 to Z0, and so on are not stored within an aplet, so if an aplet depends on the value of a global variable it's a good idea to have the program associated with Start in **VIEWS** set these values.

Scan Copyright ©  
The Museum of HP Calculators  
[www.hpmuseum.org](http://www.hpmuseum.org)

Original content used with permission.

Thank you for supporting the Museum of HP  
Calculators by purchasing this Scan!

Please do not make copies of this scan or  
make it available on file sharing services.