A Dynamic Approach to Precalculus using GeoGebra

Joseph Manthey

University of Saint Joseph

October 4, 2018

Abstract

In this session, participants will use GeoGebra to create and explore dynamic demonstrations of precalculus concepts including the reflective property of a parabola, an animation of the unit circle, transformations of sine and cosine graphs, acoustic beats and more. Be sure to bring a GeoGebra enabled device.

History of GeoGebra

GeoGebra was created by Markus Hohenwarter who began working on the project as part of his master's thesis in Computer Science and Mathematics Education at the University of Salzburg (Austria) in 2002. He designed the software to combine features of dynamic geometry software with features of computer algebra systems. Markus continued his project at Florida Atlantic University (2006–2008), Florida State University (2008–2009), and now at the University of Linz together with the help of open-source developers and translators all over the world.

Precalculus visualizations

- Odd/even functions
- Reflective property of a parabola
- Animation of the unit circle
- Transformations of trigonometric graphs
- Proof of sum and difference formulas for sine and cosine
- Acoustic beats

Some GeoGebra tips

1. **Input Bar.** We can perform constructions in GeoGebra using the Toolbar or by entering commands into the Input Bar.

Input: _____ ⟷ ⑦

2. **Sliders.** To create slider, select the Slider tool and click on the location where you wish to place the slider. When the Slider dialog box opens up, enter the name, minimum value, maximum value and the increment for the slider.
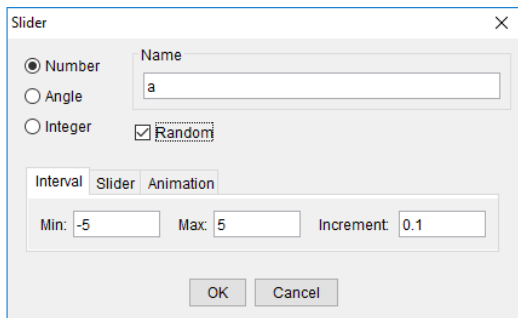
3 Buttons. To create button, select the Button tool and click on the location where you wish to place the button. When the Button dialog box opens, enter the GeoGebra code you wish to execute when the button is pushed.

OK	🗘 Button	×
	Caption: Start Animation	
	GeoGebra Script:	
	1 StartAnimation[object, true]	
	OK Cancel	

4. **Checkboxes.** To create checkbox, select the Checkbox tool and click on the location where you wish to place the checkbox. Then select the items you wish to display/hide when the checkbox is clicked.
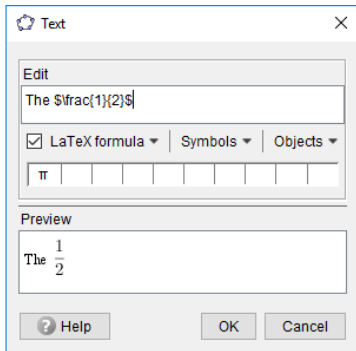
⑤ Text. To add text, select the Text tool and click on the location where you wish to place the text. You may use ordinary text and/or LaTex as needed to enter mathematical notation.

ABC		

Text ×

Edit

The $\frac{1}{2}$

☑ LaTeX formula ▾ | Symbols ▾ | Objects ▾

π									

Preview

The $\frac{1}{2}$

❓ Help   OK   Cancel

6 **Object properties.** Right-click on any object to open the Object Properties dialog box. You can then change add a caption, change the color and style of the object, set conditions for when the object is displayed, add formulas for creating dynamic colors and add GeoGebra script or Javascript.
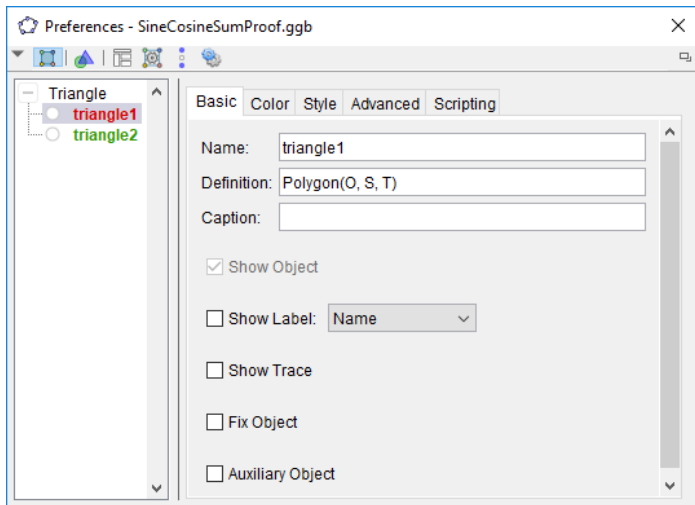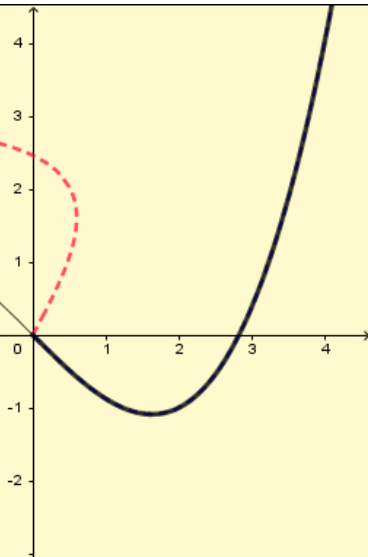
Directions

1 **Create a function.** Enter the following command into the Input Bar to create a function. Then use the Input Box tool to create an input box and link it to the function. This will permit the user to change the function.

```
f(x) = x^2 + x
```

2 **Create a copy of the right-half of the function.** Create a copy of the right half of the function which can be reflected or rotated in order to determine whether the function is odd, even or neither.
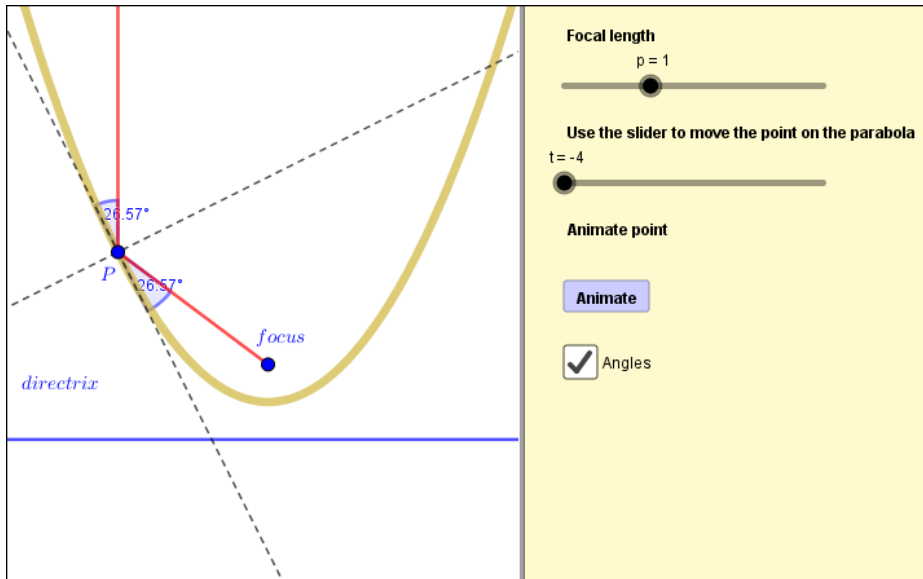
```
fr(x) = Function(f, 0, ∞)
```

3 **Test for an even function.** A function is even if it is symmetric about the y-axis. A slider will be used to create the illusion of rotating the right half of the function about the *y*-axis. Create a slider *s* with a range from -1 to 1 and an increment of 0.01. Then enter the following command into the Input Bar to create a parametric curve and attach it to the slider. As the slider is used to increase the value of s from -1 to 1, the parametric curve gives the impression of being rotated about the y-axis.

```
fle = Curve(-s t, f(t), t, 0, 20)
```

④ **Test for an odd function.** A function is odd if it is symmetric about the origin, meaning the graph is unchanged if rotated about the origin by $180°$. Create a slider α with a range from 0 to $180°$ and an increment of $1°$. Then enter the following command into the Input Bar to attach the right half of the function to the slider. The user can now use the slider to rotate the right half of the function $180°$ counterclockwise about the origin.

```
flo = Rotate(fr, α)
```

Focal length

p = 1

Use the slider to move the point on the parabola

t = -4

Animate point

Animate

☑ Angles

Parabolas have a remarkable geometric property with many applications.

Reflective property of a parabola

Rays parallel to the axis of a parabolic mirror are reflected such that they all pass through the focus. Conversely, all rays emanating from a point source placed at the focus of a parabola are parallel.

The fact that light from the focus of a parabolic mirror is reflected in a parallel beam is the basis for the design of headlights for a car. The fact that light from a distant source striking a parabolic mirror is focused at a single point is the basis for the design of solar cookers and the antennas on cell phone towers. In this section, we learn how to use GeoGebra to illustrate this property. It is worth noting that a GeoGebra applet leading to a proof would need to be designed differently. We create the parabola, focus and directrix based on the following result from precalculus.

Equation of parabola

The equation of a parabola with with focus $(0, p)$ and directrix $y = -p$ is given by

$$y = \frac{1}{4p}x^2$$
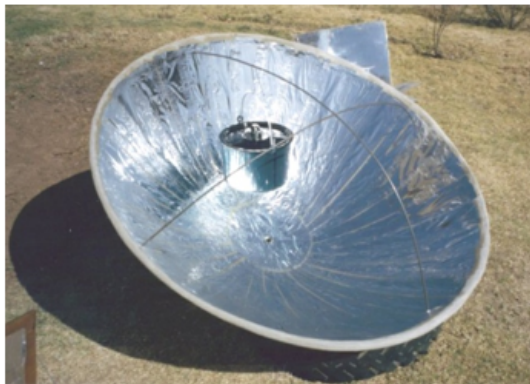
Directions

1 **Create a slider for the focal length.** Use the Slider tool to create a slider for the focal length with a range from 0 to 3 and an increment of 0.1. Set the Repeat behavior to Increasing (Once).

2 **Create a parabola and its focus and directrix.** Enter the following commands into the Input Bar to create the parabola, focus and directrix.

```
f(x) = 1/(4p) x ^ 2
F = (0, p)
y = -p
```

3 **Create a movable point on the parabola.** Create a slider for the parameter *t* with a range from -4 to 4 and an increment of 0.20. Then enter the following command into the Input Bar to create the point.

```
P = (t, f(t))
```

4 **Add line segments mimicking a ray of light entering a parabolic mirror.** It is possible to add arrows representing the direction by using Object Properties/ Decoration.

```
s1 = Segment (P , F)
s2 = Segment (( x(P), 30) , P)
```

⑤ **Use a button to animate the movable point on the parabola.** It is convenient, but not necessary to create a button to control the animation. This can be accomplished using the Button tool to create a button. The code is added by right-clicking on the button, selecting Object Properties and then the Scripting tab. The first line of the code resets the animation parameter. The next two lines select and refresh the viewing window, thereby erasing the traces. Finally, the last line of code starts the animation.

```
t = -4
SetActiveView [1]
ZoomIn [1]
SetActiveView [2]
StartAnimation [t]
```

6 **Add a tangent line and normal line.** Add a tangent line and a normal line to the parabola at point P using the following.

```
tangent = Tangent(P, f)
normal = PerpendicularLine(P, tangent)
```

7 **Measure angles of incidence and reflection.** Add points Q and R to the tangent, one on each side of point P and a point S to line segment $s2$ to assist with measuring the angles of incidence and reflection. As shown on the figure on the next slide, the angles of incidence and reflection must be defined differently when the point P is on the right half of the parabola versus the left half of the parabola (since the role of points R and Q are reversed).
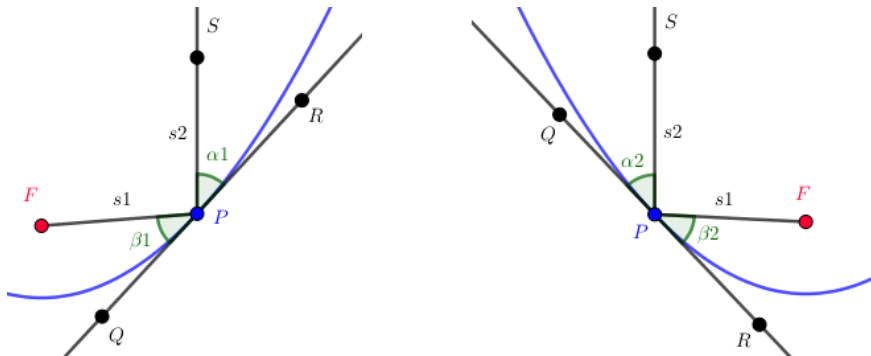
```
Q = Point(tangent)
R = Point(tangent)
S = Point(s2)
α1 = Angle(R, P, S)
α2 = Angle(S, P, Q)
β1 = Angle(F, P, Q)
β2 = Angle(R, P, F)
```

Figure 1: Measure the angles of incidence and reflection using $\alpha 1$ and $\beta 1$ when P is on the right half of the parabola (left) and using $\alpha 2$ and $\beta 2$ when P is on the left half of the parabola.

8 **Use a checkbox to control the visibility of the tangent and normal lines.** Use the Checkbox tool to create a checkbox. Right-click on the checkbox and rename it *ShowAngles*. We will use this checkbox to allow the user of the applet to decide whether or not to display the tangent line, normal line and the angles of incidence and reflection. Right-click on the tangent line, select Object Properties and click on the Advanced tab. Under the Condition to Show Object enter the Boolean variable *ShowAngles*. This will connect the tangent line to the checkbox. Repeat this process for the normal line.

```
ShowAngles
```

9 **Use the checkbox to control the visibility of the angles.** Right-click on the angle $\alpha 1$, select Object Properties and click on the Advanced tab and enter the Boolean expression shown below. The *ShowAngle* variable connect the angle $\alpha 1$ to the checkbox and the condition $x(P) \geq 0$ ensures that $\alpha 1$ is only visible when the point P is in the right-half plane. Repeat this process for the angle $\beta 1$. Repeat the process for angles $\alpha 2$ and $\beta 2$, except this time use the condition *Showangles* $\land x(P) < 0$ to ensure these angles are visible only when P is in the left-half plane.

```
ShowAngles ∧ x(P) ≥ 0
```
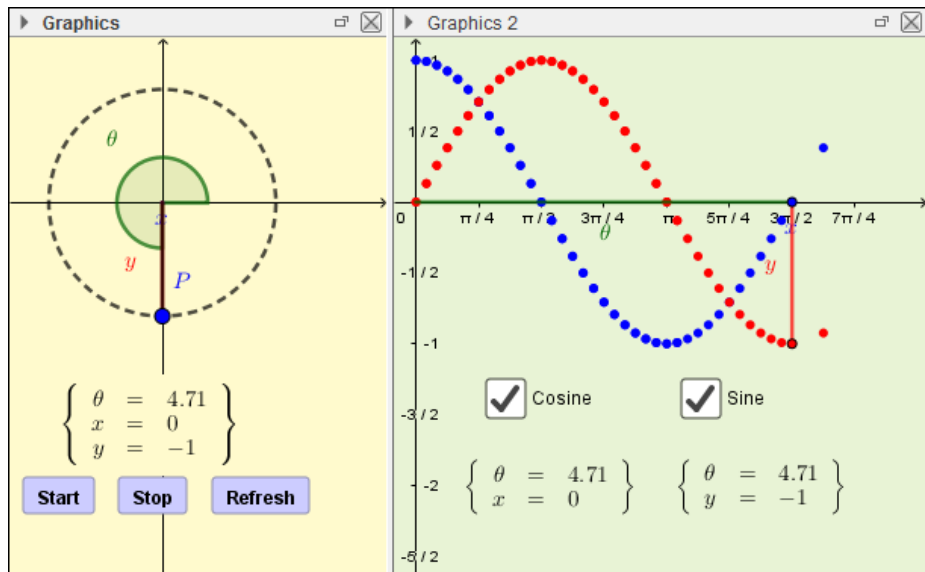
Directions for the Graphics window

GeoGebra has two graphics windows named Graphics and Graphics 2. It is convenient to display the unit circle and an embedded triangle in Graphics and the sine and cosine curves in Graphics 2.

❶ Create unit circle and a moveable point on the unit circle. The unit circle, slider and the point P on the unit circle can be created using the toolbar or by entering the following commands into the Input Bar.

```
UnitCircle: x² + y² = 1
t: Slider(0, 2 π , π/24 )
P = (cos(t), sin(t))
```

❷ Embed a triangle inside the unit circle. Define point O as the origin and point Q as the projection of point P onto the x-axis. Then create line segments for the sides of the right triangle.

```
O = (0, 0)
Q = (x(P), 0)
X = Segment(O, Q)
Y = Segment(Q, P)
H = Segment(O, P)
```

❸ Show the angle. The angle is measured relative to the x-axis.

```
R = (1, 0)
theta = Angle(R, O, P)
```

❹ Display the values of the angle, cosine and sine. To insert text, select the Text tool and then click anywhere in Graphics View. We can use LaTex commands to create mathematical notation. To display the values of the angle, cosine and sine, we click on Objects and select the object whose value we wish to display. Then when the value of the object changes, the value displayed in the text box also changes. This is called dynamic text. As you can see in the figure below, the text displays the the value of the slider t and the x and y coordinates of the point P. To obtain the x and y coordinates, we select the object P and then use the keyboard to wrap this object with $x()$ and $y()$. When doing this, be sure to stay within the small boxes as highlighted in the figure below.

Directions for the Graphics 2 window

Use View/Graphics 2 to open a second graphics window.

1. **Display the sine and cosine curves.** Enter the commands below into the Input Bar. The first two commands are used to create a segment of length t to represent the angle. The middle two commands are used to create points C, S representing the values of sine and cosine. And finally, the last two commands are used to create vertical line segments $X2$, $Y2$ representing the values of sine and cosine. Be careful to ensure these objects are displayed in the Graphics2 window, not the Graphics window. This can be accomplished by clicking the Graphics2 window prior to performing the constructions to ensure that it is the active window. It is also possible to change the graphics window after the fact by right-clicking on the object, selecting Object Properties, selecting the Advanced tab and then using the checkboxes. Turn on the trace for points C and S by right-clicking on each object and selecting Trace On.
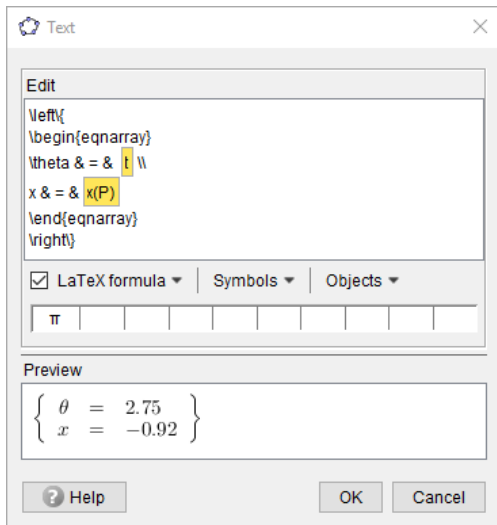
```
T = (t, 0)
theta2 = Segment(O, T)
C = (t, x(P))
S = (t, y(P))
X2 = Segment(T, C)
Y2 = Segment(T, S)
```

② **Display the values of the angle, cosine and sine using text boxes.** Create a textbox to display the value of the angle and the cosine as shown below. Then repeat this process to create a textbox for displaying the value of the angle and the sine.

③ Create checkboxes to display either the cosine, sine or both. Use the Checkbox tool to create a checkbox for the cosine. This checkbox will allow the user to show/hide the point C and segment $X2$ representing the value of the cosine and the textbox which displays the value of the cosine. Repeat this process for the sine function.
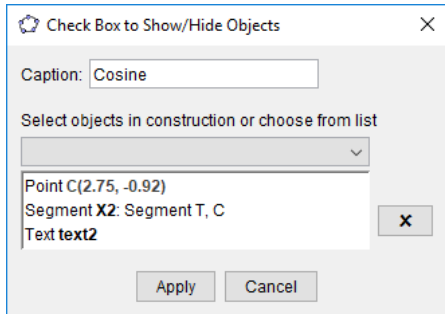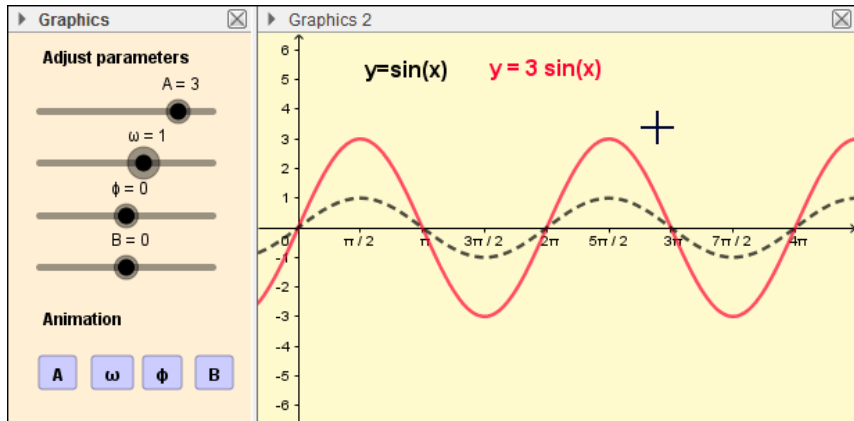
Directions for Graphics window

1. **Create buttons to control the animation.** Move back to the Graphics window and use the Button tool to create buttons for starting the animation, stopping the animation and refreshing the screen. The code for the Start and Stop buttons is self-explanatory. For the Refresh button, we use the SetActiveView and ZoomIn commands to select and refresh windows 1 and 2 (Graphics and Graphics 2).

```
//  Start animation button code
    t = 0
    StartAnimation[t, true]

//  Stop animation button code
    StartAnimation[t, false]

//  Refresh button code
    t = 0
    SetActiveView[1]
    ZoomIn[1]
    SetActiveView[2]
    ZoomIn[2]
```

Directions

Place the controls for this applet in the Graphics window and display the transformations of the sine function in the Graphics 2 window.

1. **Create the sine function and sliders.** Enter the commands shown below into the Input Bar. The first command creates the transformed sine function and GeoGebra will provide you with the option to create sliders which you can accept by pressing Create sliders. The second command creates the fundamental sine function which can be used for comparison purposes. If necessary, use the Object Properties/ Advanced tab to ensure that the slider is in the Graphics window and that the sine functions are in the Graphics 2 window.

   ```
   f(x) = A sin(ω x + φ) + B
   g(x) = sin(x)
   ```

   Right-click on each of the sliders and use the Slider tab to set the slider options as follows, other choices are possible, these reflect my personal preferences.

Slider	Min	Max	Increment
A	0	4	0.25
ω	0.25	4	0.25
ϕ	-2π	2π	π
B	-4	4	0.25

2. **Add text boxes.** Use the Insert Text tool to add a textbox to display the formula for the transformed textbox in the Graphics2 window. Use the Objects drop-down menu to select the function f. Also, this formula displays more cleanly if you do not use LaTex (Without LaTex: $y = sin(x+2)$, With LaTex: $y = 1sin(1x+2)$).

3. **Add buttons to animate the transformations.** Use buttons to animate the transformations one parameter at a time. Start by entering the following formulas into the Input Bar to create state variables for each parameter. We will program the buttons to turn off the animations when the state variables are equal to 0 and turn on the animations when the state variables are equal to 1.

```
StateA = 0
StateB = 0
Stateω = 0
Stateφ = 0
```

Use the Insert Button tool to create a button for animating the parameter A. The figure below shows the code attached to the button. The first four lines reset the parameters to the values needed for the fundamental sine function. The fifth line of code reverses the value of the variable $stateA$. If $stateA = 0$ prior to pushing the button, then it will be equal to 1 after pushing the button. If $stateA = 1$ prior to pushing the button, then it will be equal to 0 after pushing the button. This allows the button for the parameter A to turn the animation on and off. The sixth line of the code turns the animation on if $stateA = 1$ and turn the animation off if $stateA = 0$. The remaining lines of the code turn off the animations for parameters B, ω and ϕ. The buttons used to control the animations of the parameters B, ω and ϕ can be created in a similar manner.

```
A = 1
B = 0
ω = 1
ϕ = 0
StateA = 1 - StateA
StartAnimation [A, StateA == 1]
StartAnimation [B, false]
StartAnimation [ω, false]
StartAnimation [ϕ, false]
```

Deriving the sum formulas for sine and cosine

In this section, we use GeoGebra to illustrate a proof of the sum formulas for sine and cosine. We will use the proof provided by Abu al-Wafa (Persia, 940 – 988 AD). He was born in what is now Iran and moved to Bagdad when he was 20 where he made many important innovations in mathematics and astronomy. In addition to the sum and difference formulas, he is credited with discovering the law of sines for spherical triangles, the geometric solution of certain quartic equations, introducing the tangent function, creating a more accurate method for constructing tables of trigonometric functions (accurate to 8 decimal places) and the discovery of the third lunar inequality (the variation in the motion of the moon about the earth due to the effects of the sun's gravity). Abu al-Wafa was a part of the Islamic golden age, which occurred between the golden age in Greece and European renaissance.
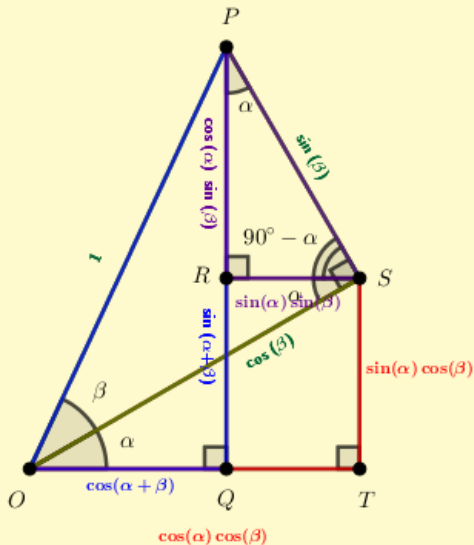
Directions for constructing the figure

There are many ways to construct the figure. One possible method is outlined below. The figure contains two right triangles with a vertex at the origin. One of the triangles has an acute angle α and the other has an acute angle β.

1. **Construct lines passing through the origin.** Construct two lines passing through the origin, one with an angle of $30°$ and the other with an angle of $65°$ relative to the positive x-axis. The values of the angles are not important and are chosen solely to provide the best appearance.

```
y1: y = tan(30°) x
y2: y = tan(65°) x
```

2. **Add the point P and two additional lines to the diagram.** Add a point P to the line $y2$ and move it into position. Then add lines $p1$ and $p2$ passing through P which are perpendicular to the x-axis and the line $y1$, respectively.

```
P = Point(y2)
p1: PerpendicularLine(P, xAxis)
p2: PerpendicularLine(P, y1)
```

❸ Locate points _Q_ and _S_. Define the point _S_ as the intersection of lines _y_1 and _p_2 and the point _Q_ as the intersection of the line _p_1 and the _x_-axis.

```
S = Intersect(y1, p2)
Q = Intersect(p1, xAxis)
```

❹ Locate points _R_ and _T_. Add a horizontal and a vertical line passing through the point _S_ and use them to locate points _R_ and _T_. Then right-click on the lines and select Show Object to hide them.

```
xs: x = x(S)
ys: y = y(S)
R = Intersect(p1, ys)
T = Intersect(xs, xAxis)
```

❺ Draw the triangles. Enter the following commands in the Input Bar to create triangles $\triangle OST$, $\triangle OPS$, $\triangle OPQ$ and $\triangle PRS$. Use Object properties to color the triangles red, green, blue and purple respectively.

```
triangle1 = Polygon(O, S, T)
triangle2 = Polygon(O, P, S)
triangle3 = Polygon(O, P, Q)
triangle4 = Polygon(P, R, S)
```

6 **Label the angles.** Use the Angle tool to label the angles in the diagram. Several of the angles will be used during the animated proof and are given the names shown below.

```
η = Angle(R, S, O)
θ = Angle(P, S, R)
κ = Angle(R, P, S)
```

7 **Label the dimensions.** Label the dimensions by entering the following commands into the Input Bar and moving the text boxes into position.

```
text1 = RotateText("1", 65°)
text2 = RotateText("\sin(β)", −60°)
text3 = RotateText("\cos(β)", 30°)
text4 = "\sin(α) \cos(β)"
text5 = "\cos(α) \cos(β)"
text6 = "\cos(α+β)"
text8 = "\sin(α) \sin(β)"
text9 = RotateText("\cos(α) \sin(β)", −90°)
```

8. **Add line segments.** Add line segments representing the key lengths used in the proof. Create line segments *OQ*, *RS* and *OT* and use Object Properties to make them a brighter color and thicker than the other segments. Also, create line segments *PR*, *ST* and *PQ*. Again, make them thicker than normal, but use a different color this time.

```
OQ: Segment(O, Q)
RS: Segment(R, S)
OT: Segment(O, T)
PR: Segment(P, R)
ST: Segment(S, T)
PQ: Segment(P, Q)
```

9. **Add text explaining the derivation.** Use the Text tool to create the text box *text*10 for displaying the derivation of the sum formula for cosine. Text boxes may contain a combination of ordinary text and LaTex. When using a combination, the LaTex code must be between dollar signs ($). Repeat this process to create text box *text*11 for displaying the derivation of the sum formula for sine.

Text ×

Edit

Sum formula for cosine

$OQ+RS=OT$

$\cos(\alpha+\beta) + \sin(\alpha) \sin(\beta) = \cos(\alpha) \cos(\beta)$
$\cos(\alpha+\beta) = \cos(\alpha) \cos(\beta) - \sin(\alpha) \sin(\beta)$

☑ LaTeX formula ▾ | Symbols ▾ | Objects ▾

| π | | | | | | | | | | |

Preview

Sum formula for cosine

$$OQ + RS = OT$$

$$\cos(\alpha + \beta) + \sin(\alpha) \sin(\beta) = \cos(\alpha) \cos(\beta)$$
$$\cos(\alpha + \beta) = \cos(\alpha) \cos(\beta) - \sin(\alpha) \sin(\beta)$$

❓ Help   OK   Cancel

Animating the proof

1. **Create a slider.** Use the Slider tool to create a slider *n* with a range between 0 and 19 with an increment of 1. We can use this slider to display the elements of the figure in a sequence to provide a proof of the sum formulas for sine and cosine.

2. **Set the timing for the display of triangle $\triangle OST$.** Consider triangle OST which we want displayed in the first step ($n = 1$) when the triangles used in the proof are initially displayed and again in the ninth step ($n = 9$) when the dimensions of the triangle are presented. Again, this can be accomplished by entering a boolean expression into the Advanced tab of the Object Properties dialog box for the triangle as shown below.
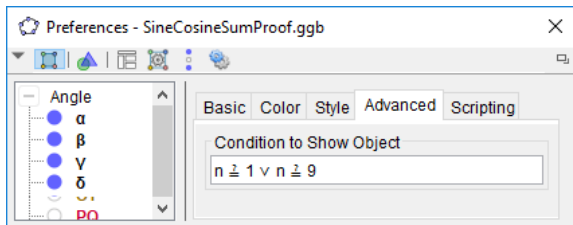
③ Set the timing for angle ∠RSO. We want some of the objects to persist after a certain step. For example, we want ∠RSO (with caption α) to persist after the fifth step ($n \geq 5$). This can be accomplished by entering a boolean expression into the Advanced tab of the Object Properties dialog box for the angle
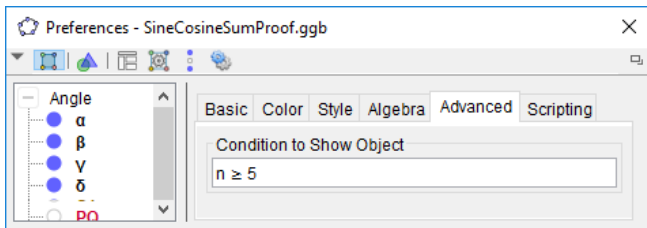
④ Set the timing for the remaining objects. The remaining objects will be attached to the slider n using the same process using the conditions to show object provided in the following table.

Object	Type	Condition	Phase
triangle1	Triangle	$n == 1 \vee n == 9$	
triangle2	Triangle	$n == 2 \vee n == 8$	Display triangles
triangle3	Triangle	$n == 3 \vee n == 10$	used in the proof.
triangle4	Triangle	$n == 4 \vee n == 11$	
η	Angle	$n >= 5$	Show how the acute
θ	Angle	$n >= 6$	angles are determined
κ	Angle	$n >= 7$	in triangle $\triangle PRS$.
text1, text2, text3	Text	$n >= 8$	
text4, text5	Text	$n >= 9$	Display the dimensions
text6, text7	Text	$n >= 10$	of the triangles
text8, text9	Text	$n >= 11$	
OQ	Segment	$n == 12$	Use the dimensions to
RS	Segment	$n == 13$	show the derivation of
OT	Segment	$n == 14$	the sum formula for
text10	Text	$n >= 15$	cosine.
PR	Segment	$n == 16$	Use the dimensions to
ST	Segment	$n == 17$	show the derivation of
PQ	Segment	$n == 18$	the sum formula for
text11	Text	$n >= 19$	sine.

Acoustic beats

When listening to two sound waves with a slightly different frequency, an alternating pattern of constructive and destructive interference creates a pattern known as acoustic beats. The frequency of the beats is equal to the absolute value of the difference between the two original frequencies. Acoustic beats can be used to tune a piano. The technique involves striking a tuning fork with the correct frequency and then adjusting the piano until the beats fade. We can model this behavior by adding $y_1 = \sin(2\pi f_1 t)$ and $y_2 = \sin(2\pi f_2 t)$ using the sum-to-product identity

$$\sin(\alpha) + \sin(\beta) = 2\sin\left(\frac{\alpha + \beta}{2}\right)\cos\left(\frac{\alpha - \beta}{2}\right)$$

to obtain

$$y = y_1 + y_2 = 2\underbrace{\sin\left[2\pi\left(\frac{f_1 + f_2}{2}\right)t\right]}_{\text{High frequency signal}}\underbrace{\cos\left[2\pi\left(\frac{f_1 - f_2}{2}\right)t\right]}_{\text{Low frequency envelope}}$$

Frequency 1 60
Frequency 2 62

Individual signals

$y_1 = \sin(120\pi t)$

$y_2 = \sin(124\pi t)$

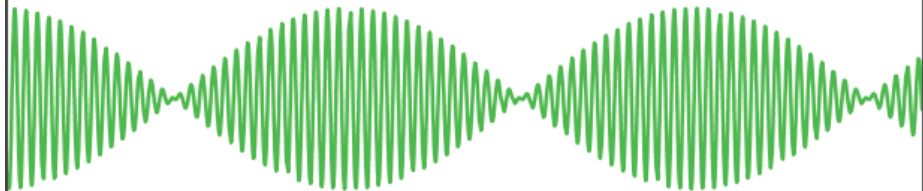**Combined signal**

$y_3(t) = y_1(t) + y_2(t)$

Play the sounds
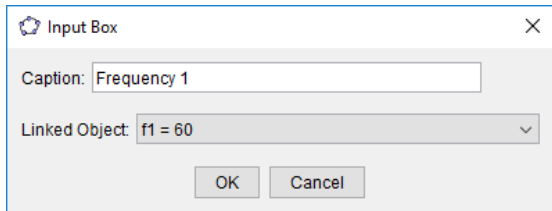
☐ y1

☐ y2

☑ y3

Play y1

Play y2

Play y3

Directions

1. Create input boxes to accept two frequencies. Start by enter the following GeoGebra commands into the Input Bar to create two frequencies. The values of the frequencies are arbitrary an can be changed later. Use the Insert Input Box tool to insert an input box that the user can enter a frequency into. The tool allows you to link the input box to one of the frequencies. Repeat this process for the second frequency.

```
f1 = 60
f2 = 61
```

2 **Graph the individual and combined signals.** Enter the following commands into the Input Bar to graph the two signals and their sum.

```
y1(x) = sin(2 π f1 x)
y2(x) = sin(2 π f2 x)
y3(x) = y1(x) + y2(x)
```

3 **Use textboxes to display the formulas of the signals.** Use the Insert Textbox tool to insert textboxes for the formulas for the two signals and their sum. Use the Objects drop-down menu to insert the frequencies so as to create dynamic text. The process is illustrated below for y_1. Repeat this process for y_2 and $y = y_1 + y_2$.
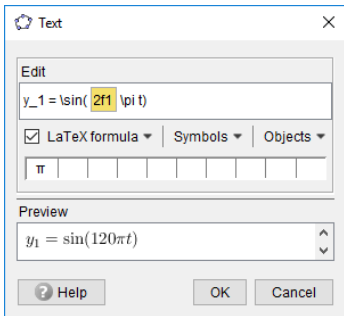
④ **Add check boxes to control which graphs are visible.** Use the Insert Check Box tool to insert check boxes to allow the user to display/hide the graphs.

⑤ **Add buttons to plays the sounds.** Use the Insert Button tool to insert buttons for playing the sounds associated with y_1, y_2 and $y = y_1 + y_2$. The GeoGebra Script associated with each button uses the PlaySound command which accepts a function and a starting and stopping time.

Resources
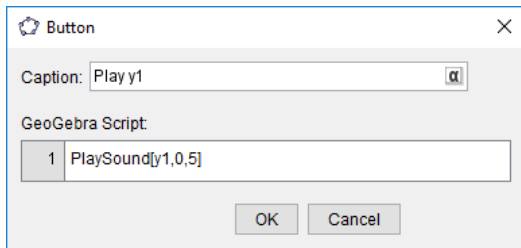
1. **GeoGebra website.** You can use GeoGebra online or download a version of GeoGebra for use on your computer from the website https://www.geogebra.org/. My personal favorite is the GeoGebra Classic 5, which can be accessed from this website via Offline Apps/App Downloads/GeoGebra Classic 5. If you click on the Classroom Resources button, you can access over 1 million GeoGebra applets. The website https://www.geogebra.org/search/Manthey will take you directly to the applets that I have created over the years.

2. **GeoGebra Tutorial and Applets.** The website http://www.malinc.se/math/ created by Maline Christersson contains an excellent GeoGebra tutorial and many outstanding GeoGebra applets.

3. **North America GeoGebra Journal.** The North American GeoGebra Journal publishes articles on a wide range of applications of GeoGebra. The articles can be accessed for free on the website https://geogebrajournal.miamioh.edu/index.php/ggbj/.

4. **Exploring Advanced Euclidean Geometry with GeoGebra.** by Gerard Venema published by the MAA introduces GeoGebra and illustrates how to use GeoGebra in Euclidean geometry.

5. **Mathematical Modeling: Applications with GeoGebra.** by Jonas Hall and Thomas Lingefjard published by Wiley shows how to use GeoGebra to explore a wide range of empirical, discrete, geometric, calculus based and differential equations based models