Jose M. Serra Jr.
Representing, Evaluating, and Graphing Functions
Objective: Analyze functions graphically, and algebraically. Observe how composite functions are related to the functions they are composed of.
Commands Used: Plot, Show, Plot Legends, PlotRange, PlotPoints, FindRoot, Reduce

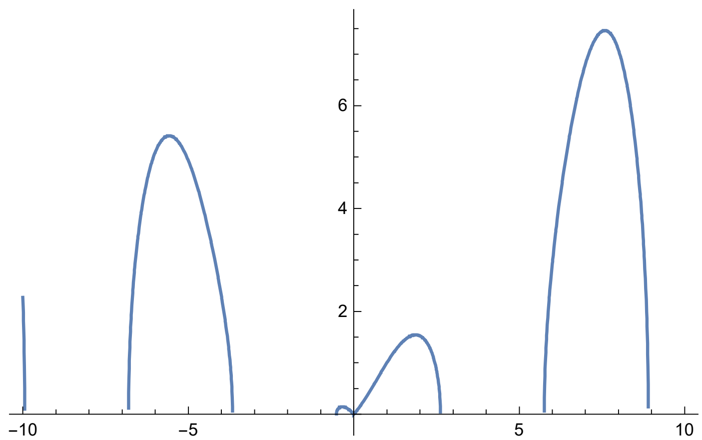
1. The function f[x]=Sqrt[x^2*Sin[x+Pi/6]] was given.

In[11]:=
```
f[x_] := Sqrt[x^2 * Sin[x + Pi / 6]]
```

The function was then graphed with a domain restriction from -10 to 10.

In[12]:=
```
Plot[f[x], {x, -10, 10}]
```

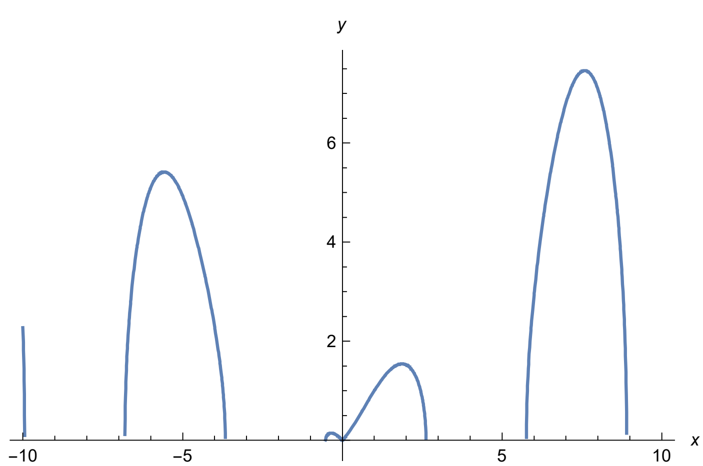
Out[12]=



The plotted function then had its axises labelled x, and y.

In[13]:=
```
Show[%12, AxesLabel → {HoldForm[x], HoldForm[y]},
 PlotLabel → None, LabelStyle → {GrayLevel[0]}]
```

Out[13]=



(a) The function appears to have x-intercepts in the graph.
The function is then set equal to zero, and solved.

In[14]:= `Solve[f[x] == 0]`

Out[14]= $\left\{\{x \to 0\}, \left\{x \to -\frac{\pi}{6}\right\}\right\}$

Only two zeroes seem to appear from the solve command being used to determine when f[x] is equal to zero.

(b) The solve feature solves for only a couple of zeroes with in the domain of the function, but not all the zeroes throughout the whole function. Reading the output of the Solve function it reads that at x=0 there is a zero, and at x=Pi/6 there is a zero too.

Shifting through the catalogue of commands, I found an “expanded” Solve command which could solve for every zero within the domain, and tried two times to achieve the desired output of every zero in the domain.

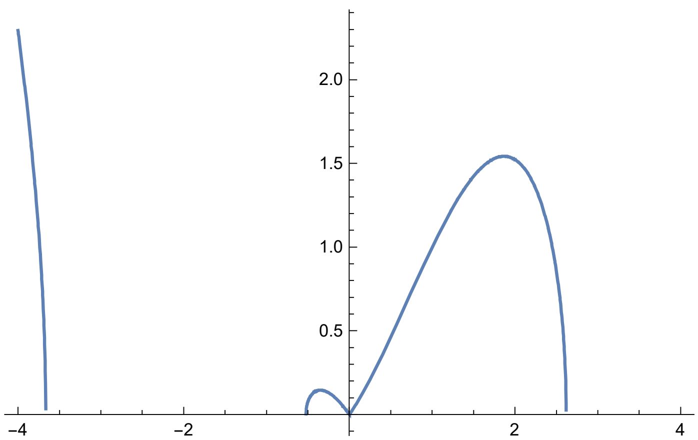
In[17]:= `Solve[f[x] == 0, x, Reals]`

Out[17]= $\left\{\{x \to 0\}, \left\{x \to \text{ConditionalExpression}\left[-\frac{\pi}{6} + 2\pi C[1], C[1] \in \text{Integers}\right]\right\}, \left\{x \to \text{ConditionalExpression}\left[\frac{5\pi}{6} + 2\pi C[1], C[1] \in \text{Integers}\right]\right\}\right\}$

The function f[x] was graphed with a domain restriction from -4 to 4.

In[18]:= `Plot[f[x], {x, -4, 4}]`

Out[18]=



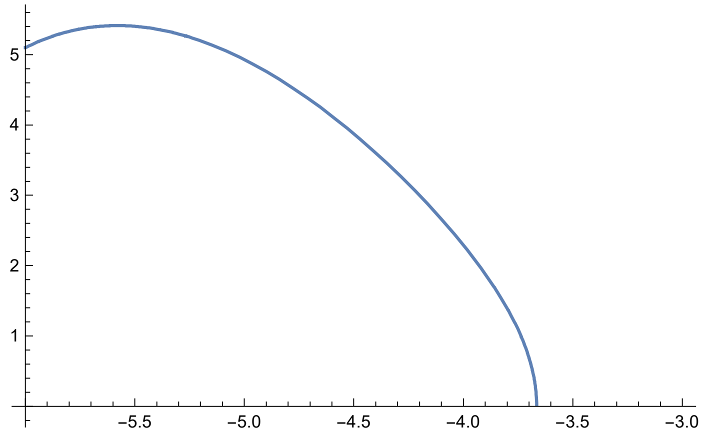
Within this view it appears as if there are only two zeroes in this view even though it appeared in the previous view that there were more, and also it’s noted that the function isn’t fully graphed at the points that were appeared to be zeroes.

(c) Further examination of the function, under the new domain restriction reveals that the graph does not intercept the x-axis at the point x=(5*Pi)/6 , or at x=-(7*Pi)/6 which are zeroes as revealed by the expanded solve in input 17. (Work for x=(5*Pi)/6 and x=-(7*Pi)/6 => Sin(x+Pi/6)=0 therefore, x+Pi/6=0 and x+Pi/6= Pi hence, x=-Pi/6+2*Pi*Z and x=(5*Pi)/6+2*Pi*Z Z is in the domain of Integers. If Z is set to -1 then x=(5*Pi)/6+2*Pi*(-1) => x=-(7*Pi)/6. However, the graph appears have zeroes at x=0, and x=-pi/6.

Afterwards, the function f[x] was then graphed with a domain restriction from -6 to -3.

In[19]:= `Plot[f[x], {x, -6, -3}]`

Out[19]=



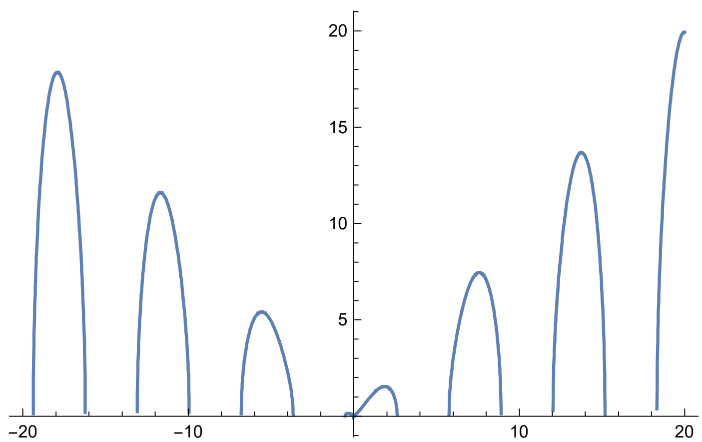
The function in this view shows the graph intercept the x-axis at x=-(7Pi/6).

(d) The graph within this domain restriction appears to have to have a zero at x=-(7*pi)/6, unlike the other views which have been shown.

The function is then graphed instead of a domain restriction of -6 to -03, but -20, to 20.

In[20]:= `Plot[f[x], {x, -20, 20}]`

Out[20]=



This view shows more crests in the graph, and shows more zeroes; however, there seems to be a overall problem with the way Mathematica graphs because it cuts out some of the zeroes.
(d) There appears to be a many zeroes in the graph this time in this view.
Overall this problem shows that every time something is graphed, to completely see or analyze the graph the view from which the function is being viewed must be changed in order to visually better see the function. At the end it was the zeroes were seen, but the views had to be changed in order to better view the zeroes. The solve command is also tricky because it outputted only a couple of solutions; however, the expanded solve feature which I discovered gives all the zeroes that the function has.

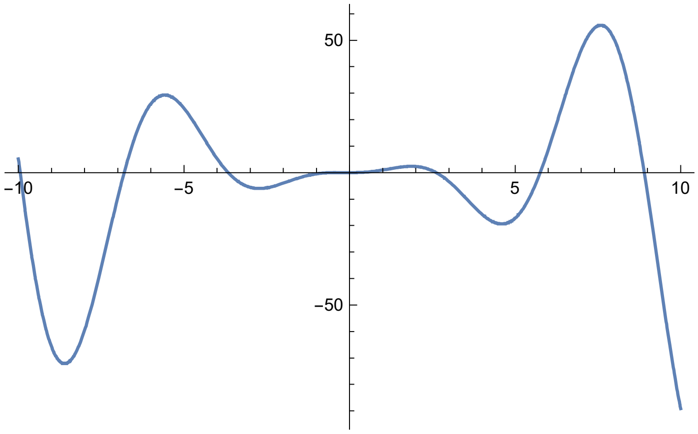
2. Given the function g[x]:=x∧2*Sin[x+Pi/6]

In[21]:= `g[x_] := x^2 * Sin[x + Pi / 6]`

The function g[x] was then plotted on the the interval from -10, to 10.

In[23]:=
```
Plot[g[x], {x, -10, 10}]
```

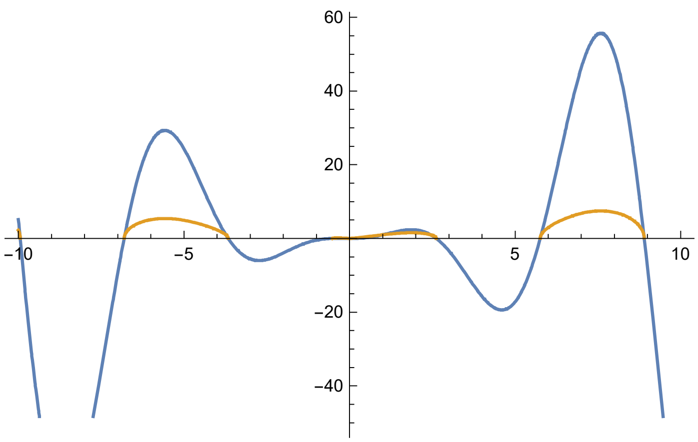
Out[23]=



This function appears similar to the f[x], since they seem to share the same appearance.
g[x], and f[x] are both plotted.

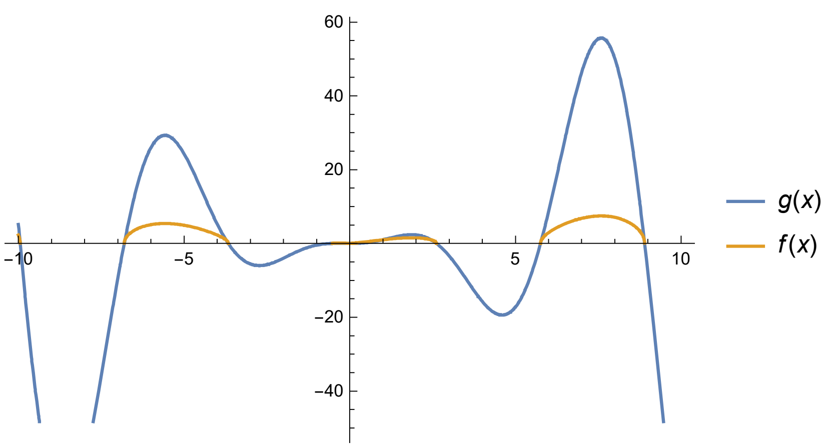
In[24]:=
```
Plot[{g[x], f[x]}, {x, -10, 10}]
```

Out[24]=



In[25]:=
```
Plot[{g[x], f[x]}, {x, -10, 10}, PlotLegends → "Expressions"]
```

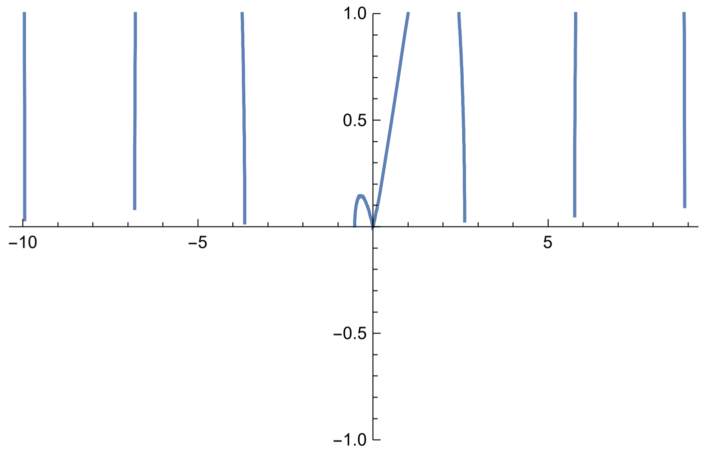
Out[25]=



The both function seemingly appear to have both the same zeroes, and both functions have been labeled in the graph.
(a) The function g[x], and f[x] has both the same zeroes because f[x]=h[g[x]]. h[u] is equal to Sqrt[u],

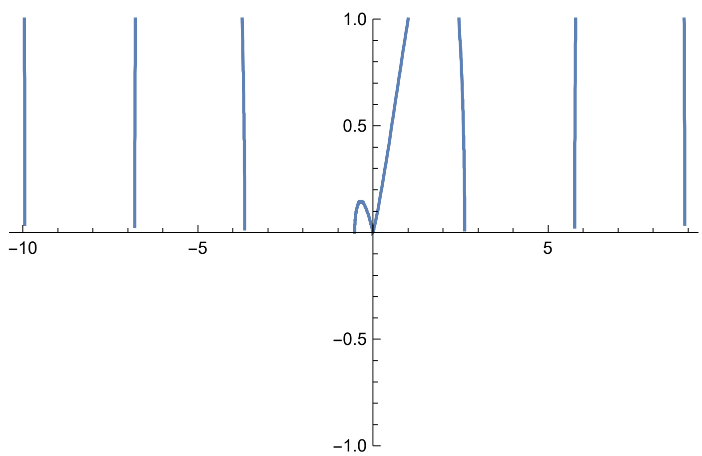
and u is equal to g[x] which is x^2*Sin[x+Pi/6] so each time g[x] is equal to 0, then f[x] is equal to zero.
(b)Taking the same approach, from 2(a), which is f[x]=h[g[x]]. h[u] is equal to Sqrt[u], and u is equal to g[x] which is x^2*Sin[x+Pi/6] every time u is negative that region is undefined for f(x).
The function f[x] was plotted with a domain restriction of -10, to 10, and a range restriction of -1, to 1. There were 100 points used in this plot.

In[26]:=
```
Plot[f[x], {x, -10, 10}, PlotRange → {-1, 1}, PlotPoints → 100]
```

Out[26]=


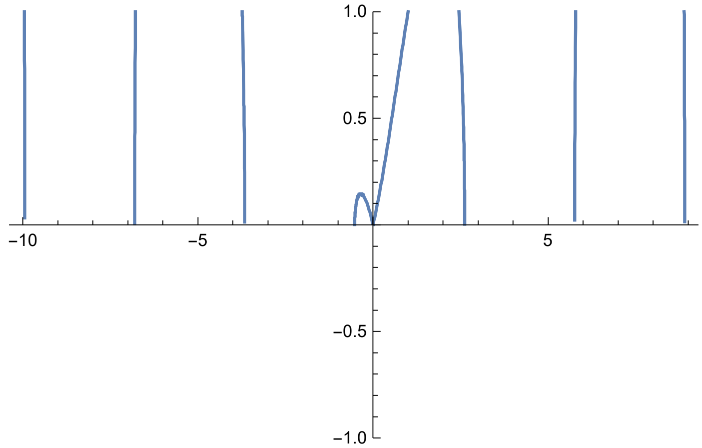
Where the function should have zeroes there appears to be no evident zero present.
The function f[x] was plotted again with a domain restriction of -10, to 10, and a range restriction of -1, to 1. In this plot 500 points were used.

In[27]:=
```
Plot[f[x], {x, -10, 10}, PlotRange → {-1, 1}, PlotPoints → 500]
```

Out[27]=


It appears that as the number of plotted points increases there function seems to approach the zeroes it should have at those points.
The function f[x] was plotted once more with a domain restriction of -10, to 10, and a range restriction of -1, to 1. However, in this plot 1000 points were used to plot this graph.

In[28]:=
```
Plot[f[x], {x, -10, 10}, PlotRange → {-1, 1}, PlotPoints → 1000]
```

Out[28]=


The function using 1000 gives a more detailed view of the function, and the zeroes which its supposed to have.
(c)The lines of the graph are getting closer and closer to the x-axis, thus revealing the possible x-intercepts. The x-intercepts do exist, except that increasing the number of PlotPoints the zeroes will be revealed.
This exercise overall revealed that an analysis of the inner function can make plotting the composite function easier to plot the function. Another lesson learned was that the in the specific function which was used in this exercise had the same zeroes as the inner function (i.e g[x]=x^2*Sin[x+Pi/6], and f(x)=Sqrt[g[x]]). The essential lesson learned was that there is a way to increase the accuracy of the plots within Mathematica.

The roots for f[x]=Sqrt[x∧2*Sin[x+Pi/6]], in the interval of -0.6, to -0.4, were found

In[29]:=
```
FindRoot[f[x] == 0, {x, -0.6, -0.4}]
```

Out[29]= $\{x \to -0.523599 + 5.26029 \times 10^{-16}\, i\}$

The root appears to have an imaginary component which I believe the root from g[x]= x^2*Sin[x+Pi/6] will not have.
The roots for g[x]=x^2*Sin[x+Pi/6], in the interval of -0.6, to -0.4 were found

In[31]:=
```
FindRoot[g[x] == 0, {x, -0.6, -0.4}]
```

Out[31]= $\{x \to -0.523599\}$

As analyzed the answer does not have the imaginary component as the the first root have.
(d) These commands should both have the same answer. The first command must have taken a different route in analyzing the problem, and the second one did. This conclusion comes from how f[x] is Sqrt[x∧2*Sin[x+Pi/6]], and g[x] is x^2*Sin[x+Pi/6].

The possible roots were find using the reduce command.

In[32]:=
```
Reduce[f[x] == 0, x]
```

Out[32]= $\left(C[1] \in \text{Integers} \,\&\&\, \left(x == -\frac{\pi}{6} + 2\pi C[1] \;||\; x == \frac{5\pi}{6} + 2\pi C[1]\right)\right) \;||\; x == 0$

(e) Reduce gives the possible roots in a clear cut manner solving for the all the possible roots each term yields.

In conclusion, this exercise helped instruct how to view accurately a graph by enhancing it using the PlotPoints command. This command allowed more points to be plotted which allowed further access to how the function actually looked like if all the points were plotted, thus removing a bit of the ambiguity, of where its not defined. The find root feature gives the root but the steps taken to get the root could vary, therefore, altering the solution slightly as in the case output 29 which had a miniscule imaginary component. The reduce feature is useful since it gives all the possible zeroes the function may have.

3. Completing the square in for the function m(x)=1-2x-x^2. The first step taken would be to reorder the terms and to factor out the negative sign for only the terms that have x therefore yielding m(x)=-(x^2+2x)+1. Next, the step would be to add 1 inside the parentheses, and outside the parentheses, yielding m(x)=-(x^2+2x+1)+1+1. Finally factoring and combining the terms m(x)=-((x+1)^2)+2, and then simplifying yields m(x)=-(x+1)^2+2. The equation was taken from the form m(x)=ax^2+bx+c, to m(x)=a(x-h)^2+k

(a) The coefficient a is equal to -1, h is equal to -1, and k is equal to 2.

(b)

The coefficient a was defined.

In[34]:=
```
a := -1
```

The coefficient h was defined.

In[35]:=
```
h := -1
```

The coefficient k was defined.

In[36]:=
```
k := 2
```

Function m1[x] was defined as x^2

In[33]:=
```
m1[x_] = x^2
```

Out[33]= x^2

The function n1[x] was defined as -x^2.

In[39]:=
```
n1[x_] := -x^2
```

The function n2[x] was defined as (x-(-1))^2

In[40]:=
```
n2[x_] := (x - (-1))^2
```

The function n3[x] was defined as x^2+2.

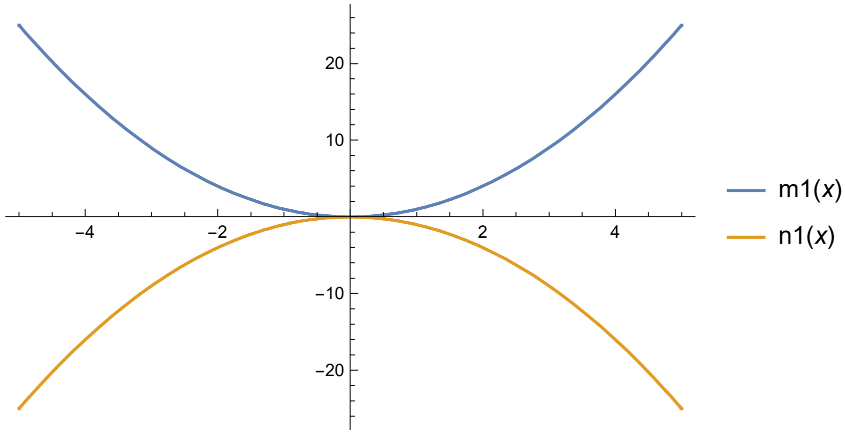
In[41]:=
```
n3[x_] := x^2 + 2
```

The function n4[x] was defined as -(x-(-1))^2+2.

In[42]:=
```
n4[x_] := -(x - (-1))^2 + 2
```

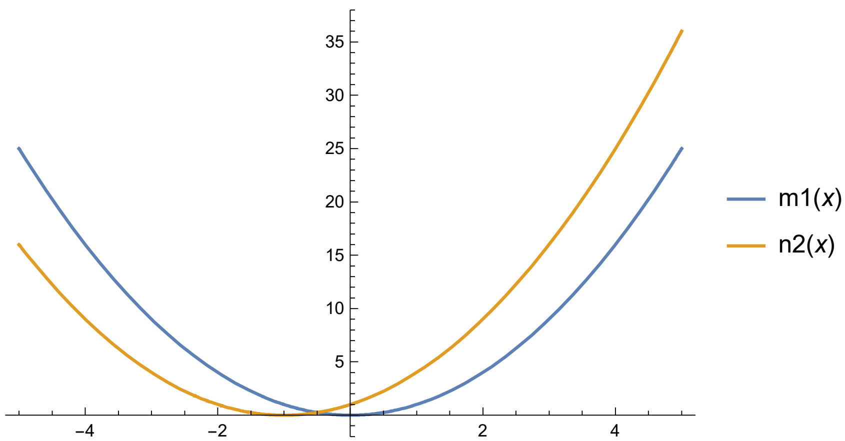
(b) The function m1[x], is graphed besides the function n1[x].

In[43]:= `Plot[{m1[x], n1[x]}, {x, -5, 5}, PlotLegends → "Expressions"]`

Out[43]=


(c) The coefficient a, being -1, causes a reflection across the x-axis in comparison to the parent function.
(b) The function m1[x], is graphed besides the function n2[x].

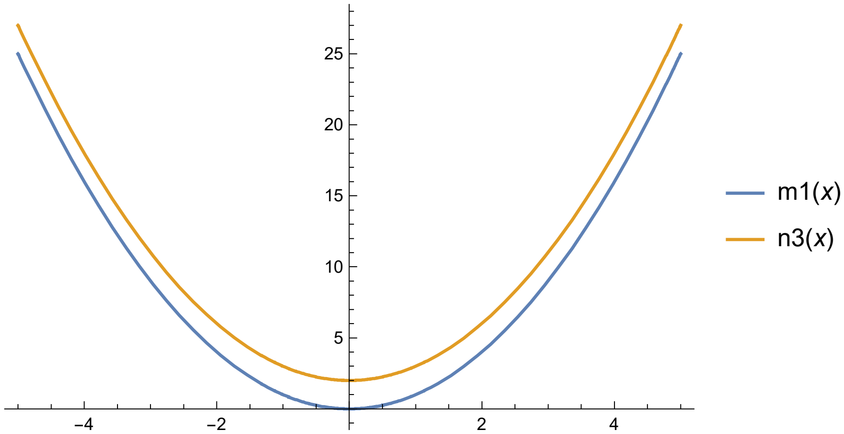
In[44]:= `Plot[{m1[x], n2[x]}, {x, -5, 5}, PlotLegends → "Expressions"]`

Out[44]=


(c) The function having h equal to -1, entails that there is a horizontal shift of 1 unit to the left in comparison to the parent function.
(b) The function m1[x], is graphed besides the function n3[x].

In[45]:=
```
Plot[{m1[x], n3[x]}, {x, -5, 5}, PlotLegends → "Expressions"]
```

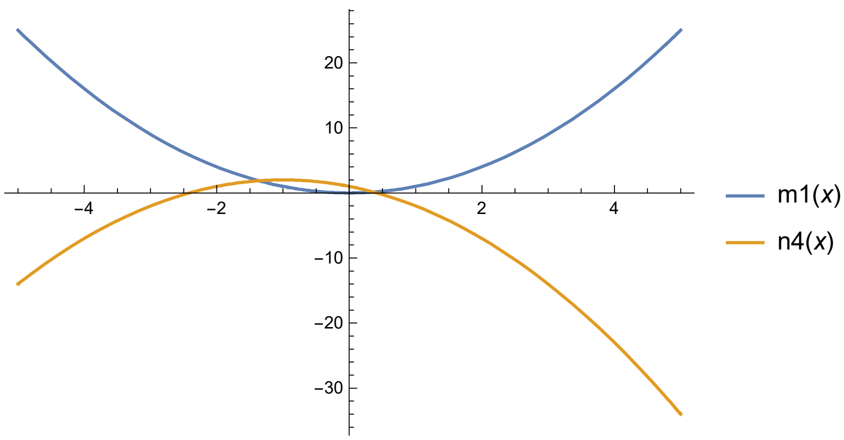
Out[45]=



(c) k is responsible for the vertical shifts, and since its equal to 2 there is a vertical shift up 2 units in comparison to the parent function.
(b) The function m1[x], is graphed besides the function n4[x].

In[46]:=
```
Plot[{m1[x], n4[x]}, {x, -5, 5}, PlotLegends → "Expressions"]
```

Out[46]=



(b) and (c) In b the graphs transformations were individually picked out and plotted next to the parent functions. Each graph plotted shows the various transformations which n4[x] under went in comparison to the parent function. Viewing each separate component shows how a,h, and k affect the function. It is shown that a is responsible for the reflections, and vertical stretches, and h is responsible for the horizontal shifts, and k the vertical shifts. The function having all those transformations strung together yields a parabola opening downward, horizontally shifted 1 unit to the left, and vertically shifted 2 units up.
(d)In comparison m1[x] versus n1[x] the roots are the same since there graph has its vertex at (0,0); however, the maximum for n1[x] is at (0,0), meanwhile the minimum for m1[x] is at (0,0).
(e)For the comparison among m1[x], and n2[x], m1[x] has its minimum at (0,0), and n2[x] has its minimum at (-1,0).
(f)As a comparison is made among m1[x], and n[3], m1[x] has its minimum at (0,0), and n[3] has its minimum at (0,2)
(g)The graph m1[x] has its root at (0,0), just as its minimum is at (0,0), and the n4[x] has its roots at x=-1+Sqrt[2], and x=-1-Sqrt[2], and its maximum at (-1,2).

In[47]:=
```
Reduce[n4[x] == 0, x]
```

Out[47]= $x == -1-\sqrt{2} \;||\; x == -1+\sqrt{2}$

Used to find the roots of n4[x].

This project gave useful insight on the transformations of functions, as well as composite functions. The most useful asset of this project was the experience with Mathematica and its various basic functions. These functions can be used to analyze more complicated functions, and can help figure out why there might be some inaccuracies in Mathematica.