Solving the Schrodinger Equation with linear potential.

```
se = (-c f''[x] + (b x - e) f[x]) == 0
```

$$(x\,b - e)\,f[x] - c\,f''[x] == 0$$

We can solve this with *Mathematica*'s built in differential equation solver

```
DSolve[se, f[x], x]
```

$$\left\{\left\{f[x] \to \text{AiryBi}\left[\left(\frac{b}{c}\right)^{1/3}\left(x - \frac{e}{b}\right)\right] C[1] + \text{AiryAi}\left[\left(\frac{b}{c}\right)^{1/3}\left(x - \frac{e}{b}\right)\right] C[2]\right\}\right\}$$

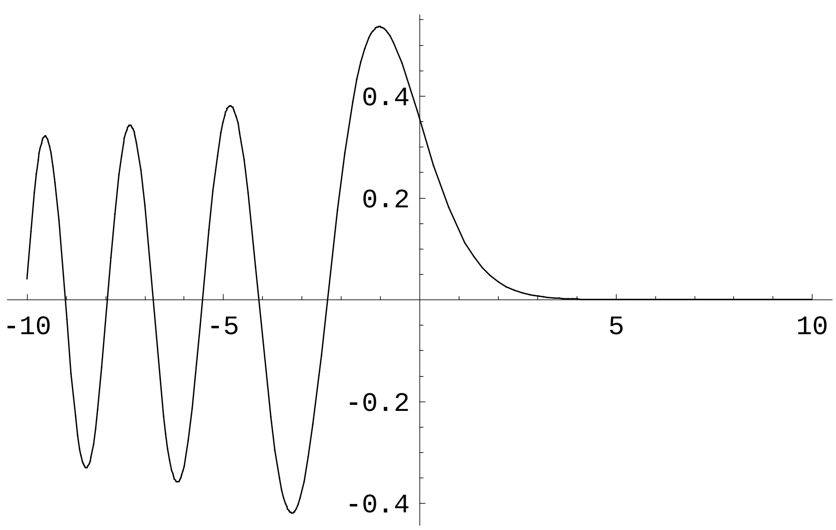
We get two solutions, Ai[x] and Bi[x]. Ai(x)-> 0 as x→∞ while Bi(x)→ ∞ as x→∞. The Ai[x] is the desired solution. (i.e C[1] = 0)

$$y[x] = c2 * \text{AiryAi}\left[\left(-\frac{b}{c}\right)^{1/3}\left(x - \frac{e}{b}\right)\right]$$

$$c2\,\text{AiryAi}\left[\left(-\frac{b}{c}\right)^{1/3}\left(x - \frac{e}{b}\right)\right]$$

Here's a plot of the Ai[x] function.

```
ap = Plot[AiryAi[x], {x, -10, 10}, PlotRange -> All];
Display["airyplot.TIFF", %, "TIFF"];
```



To find the bound states, we need to numerically search for the roots of the function. (i.e. where it's equal to zero). Here's the first one.

```
FindRoot[AiryAi[x] == 0, {x, -3}]
```

```
{x Æ -2.33811}
```

Let's tabulate the roots we find using different starting points along the x axis.

```
roots = Table[FindRoot[AiryAi[x] == 0, {x, -i}], {i, 20}]
```

```
{{x Æ 17.541}, {x Æ -2.33811}, {x Æ -2.33811}, {x Æ -4.08795},
 {x Æ -4.08795}, {x Æ -5.52056}, {x Æ -6.78671}, {x Æ -7.94413},
 {x Æ -9.02265}, {x Æ -10.0402}, {x Æ -11.0085}, {x Æ -11.936},
 {x Æ -12.8288}, {x Æ -13.6915}, {x Æ -16.1327}, {x Æ -16.1327},
 {x Æ -16.9056}, {x Æ -16.1327}, {x Æ -19.1264}, {x Æ -19.8381}}
```

Let's clean thing up a bit and put things into order. Note the last one correponds to the "root" at +∞. It's just that we ran out of iterations before we got there. The negative roots are the ones we want.

```
x /. roots // Union
```

```
{-10.0402, -9.02265, -7.94413, -6.78671, -5.52056,
 -4.08795, -4.08795, -2.33811, -2.33811, 17.541}
```

```
roots = {-10.04017434152352892`,
  -9.0226508533408527`, -7.94413358739371666`,
  -6.786708090071631`, -5.52055973421671364`,
  -4.08794944413097027`, -2.33810741637309948`}
```

```
{-10.0402, -9.02265, -7.94413,
 -6.78671, -5.52056, -4.08795, -2.33811}
```

■ Energy Levels

Just as with the 1-D box, the wavefunction must vanish at x=0,. Thus, we can write,

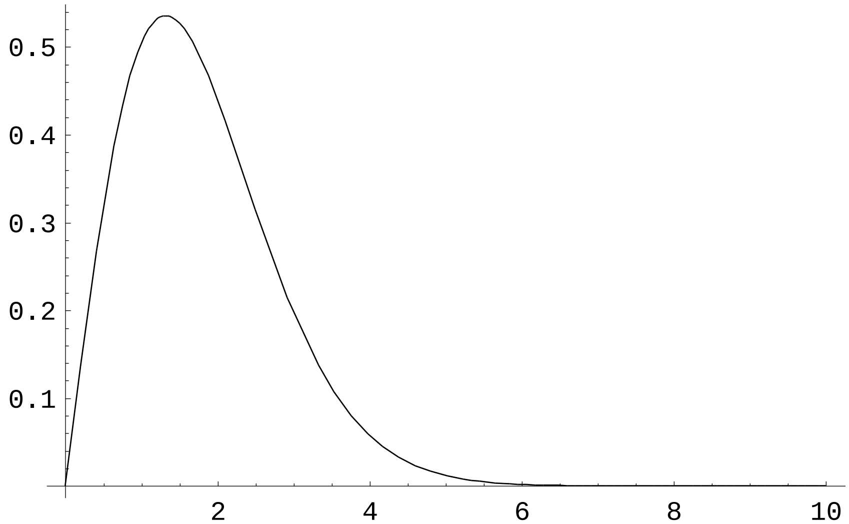
$$\texttt{e1 = Solve}\left[\left(-\frac{b}{c}\right)^{1/3}\left(\frac{e}{b}\right) == 2.33810741637309948\text{`},\ e\right]$$

$$\left\{\left\{e \rightarrow \frac{2.33811\, b}{\left(-\frac{b}{c}\right)^{1/3}}\right\}\right\}$$

```
y[x] /. e1 /. {b -> 1, c -> -1, c2 -> 1}
```

```
{AiryAi[-2.33811 + x]}
```

```
Plot[AiryAi[-2.33810741637309948` + x],
 {x, 0, 10}, PlotRange -> All]
```



Ö Graphics Ö

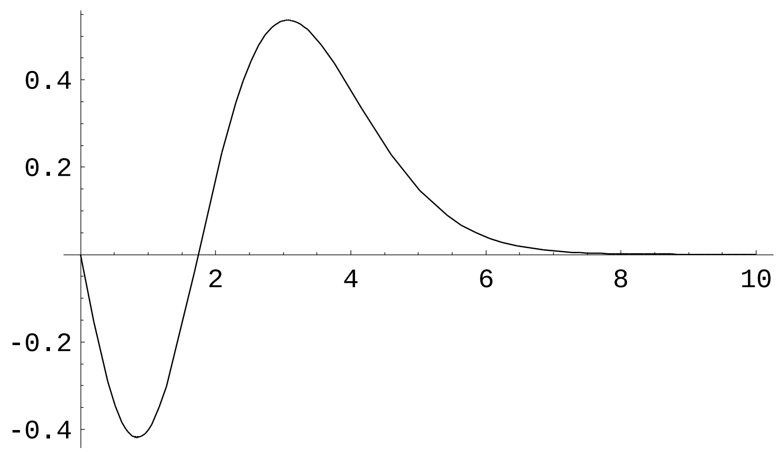
e2 = Solve$\left[\left(-\frac{b}{c}\right)^{1/3}\left(\frac{e}{b}\right) == 4.08794944413097027`, e\right]$

$\left\{\left\{e \to \frac{4.08795\, b}{\left(-\frac{b}{c}\right)^{1/3}}\right\}\right\}$

```
y[x] /. e2 /. {b -> 1, c -> -1, c2 -> 1}
```

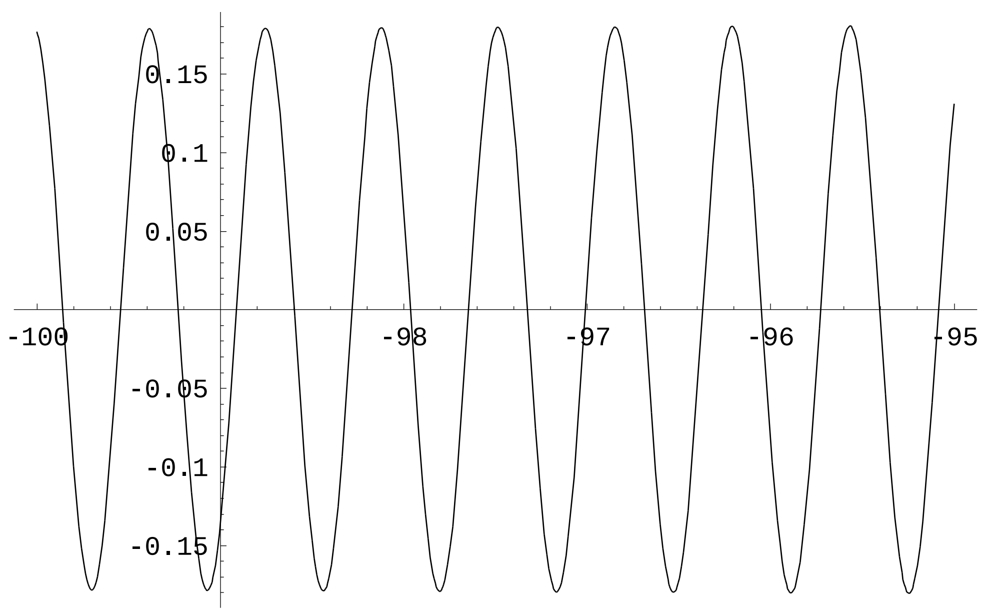
{AiryAi[-4.08795 + x]}

```
Plot[AiryAi[-4.08794944413097027` + x],
 {x, 0, 10}, PlotRange -> All]
```



Ö Graphics Ö

```
Plot[AiryAi[x], {x, -100, -95}, PlotRange -> All]
```



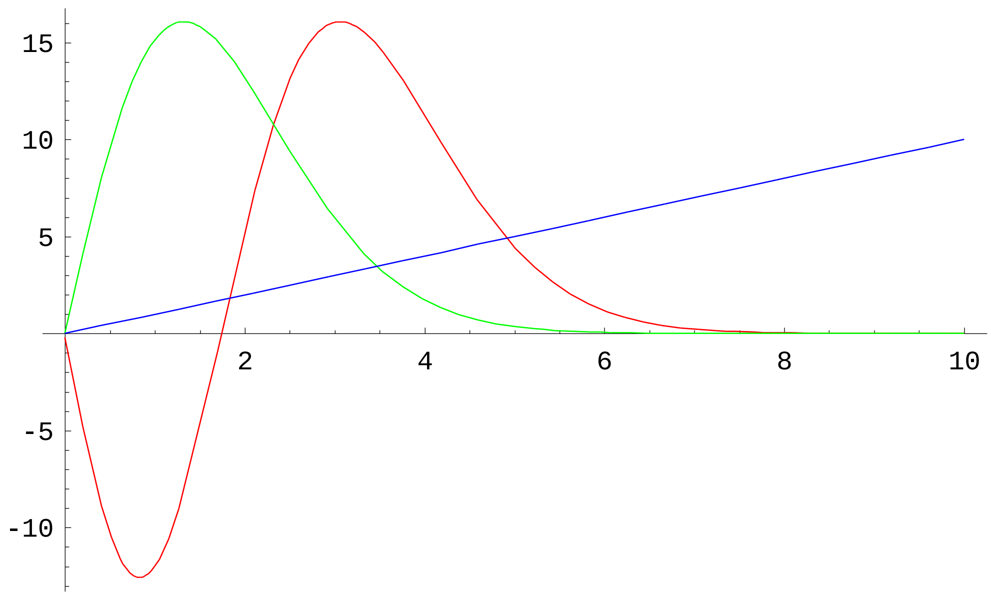
Ö Graphics Ö

Asymptotically, Ai[x] becomes a sine function.

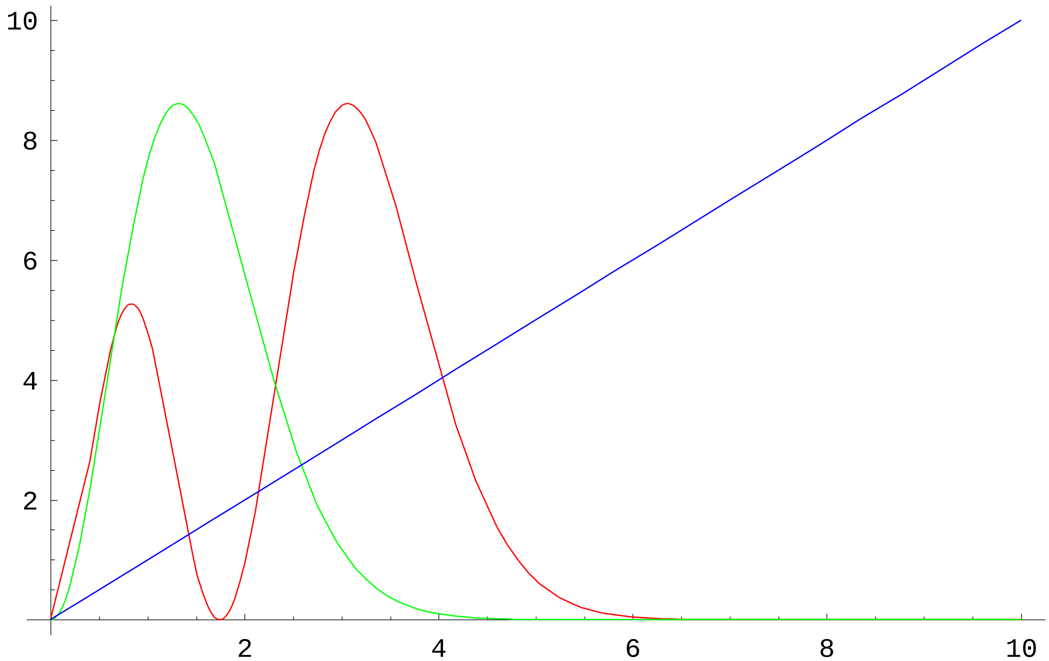
■ Wave functions

Finally, to see how things look, let's put everything into "reduced" units and plot the wavefunction and the potential on the same plot (scaling the wavefunction to make it look good.)

```
plot2 = Plot[{30 * AiryAi[(x - 4.08)], 30 * AiryAi[(x - 2.3381)], x },
   {x, 0, 10}, PlotRange -> All,
PlotStyle -> {RGBColor[1, 0, 0], RGBColor[0, 1, 0],
      RGBColor[0, 0, 1]}];
```



```
plot2 =
  Plot[{30*AiryAi[(x - 4.08)]^2, 30*AiryAi[(x - 2.3381)]^2, x},
   {x, 0, 10}, PlotRange -> All,
PlotStyle -> {RGBColor[1, 0, 0], RGBColor[0, 1, 0],
     RGBColor[0, 0, 1]}];
```



Basically the same plot as before, except that we're plotting the probability distrubution rather than the wavefunction amplitudes. Note how there is a finite probability that the wavefunction can penertrate into the classically forbidden regions.