

3.016 Problem Set #2, 2010

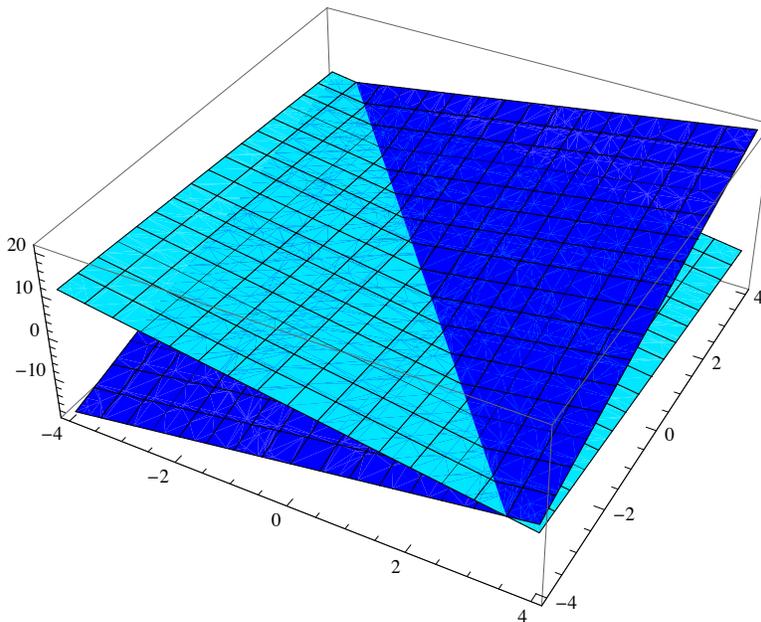
I2-1

Define plane 1 and 2,

```
plane1 = z == 1 + 2 x + 3 y;  
plane2 = y == 2 - 2 z - 4 x;
```

Plot the two planes using Plot3D,

```
planesPlot =  
Plot3D[{1 + 2 x + 3 y, 1 - y / 2 - 2 x}, {x, -4, 4}, {y, -4, 4}, PlotStyle -> {Blue, Cyan}]
```



Solve for the intersection line in terms of z,

```
lineSol = Solve[{plane1, plane2}, {x, y}]
```

```
{ {x -> -7/10 (-1 + z), y -> 4/5 (-1 + z)} }
```

The parametric form of the intersection line in terms of z,

```
eqline = {x, y, z} /. lineSol[[1]]
```

```
{ -7/10 (-1 + z), 4/5 (-1 + z), z }
```

```
ppoint0 = {7/10, -4/5, 0}
```

```
{ 7/10, -4/5, 0 }
```

`lineVector = {-7 / 10, 4 / 5, 1}`

$$\left\{-\frac{7}{10}, \frac{4}{5}, 1\right\}$$

`tt = lineVector / Norm[lineVector]`

$$\left\{-\frac{7}{\sqrt{213}}, \frac{8}{\sqrt{213}}, \frac{10}{\sqrt{213}}\right\}$$

The parametric form of the line can be written as `ppoint0 + tt*z`

`pp = {x, y, z}; zero = {0, 0, 0};`

`ww = pp - ppoint0`

$$\left\{-\frac{7}{10} + x, \frac{4}{5} + y, z\right\}$$

`dd1 = ww - (ww.tt) tt`

$$\left\{-\frac{7}{10} + x + \frac{7 \left(-\frac{7 \left(-\frac{7}{10} + x \right)}{\sqrt{213}} + \frac{8 \left(\frac{4}{5} + y \right)}{\sqrt{213}} + \frac{10 z}{\sqrt{213}} \right)}{\sqrt{213}}, \right. \\ \left. \frac{4}{5} + y - \frac{8 \left(-\frac{7 \left(-\frac{7}{10} + x \right)}{\sqrt{213}} + \frac{8 \left(\frac{4}{5} + y \right)}{\sqrt{213}} + \frac{10 z}{\sqrt{213}} \right)}{\sqrt{213}}, z - \frac{10 \left(-\frac{7 \left(-\frac{7}{10} + x \right)}{\sqrt{213}} + \frac{8 \left(\frac{4}{5} + y \right)}{\sqrt{213}} + \frac{10 z}{\sqrt{213}} \right)}{\sqrt{213}} \right\}$$

dd is the distance² from point {x, y, z} to the line `ppoint0 + tt*z`

`dd = Simplify[SquaredEuclideanDistance[dd1, zero], x ∈ Reals && y ∈ Reals && z ∈ Reals]`

$$\frac{1}{213} (164 x^2 + 149 y^2 - 160 y (-1 + z) + 113 (-1 + z)^2 + 28 x (-5 + 4 y + 5 z))$$

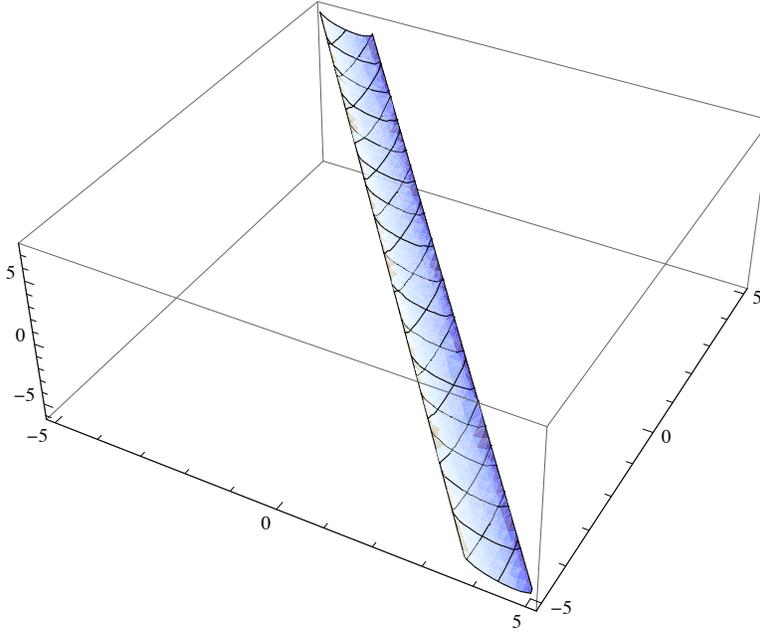
$\frac{1}{213} (164 x^2 + 149 y^2 - 160 y (-1 + z) + 113 (-1 + z)^2 + 28 x (-5 + 4 y + 5 z)) = \frac{1}{4}$ would be the equation describing the desired cylinder.

`solveCylin = Solve[dd == 1 / 4, {z}]`

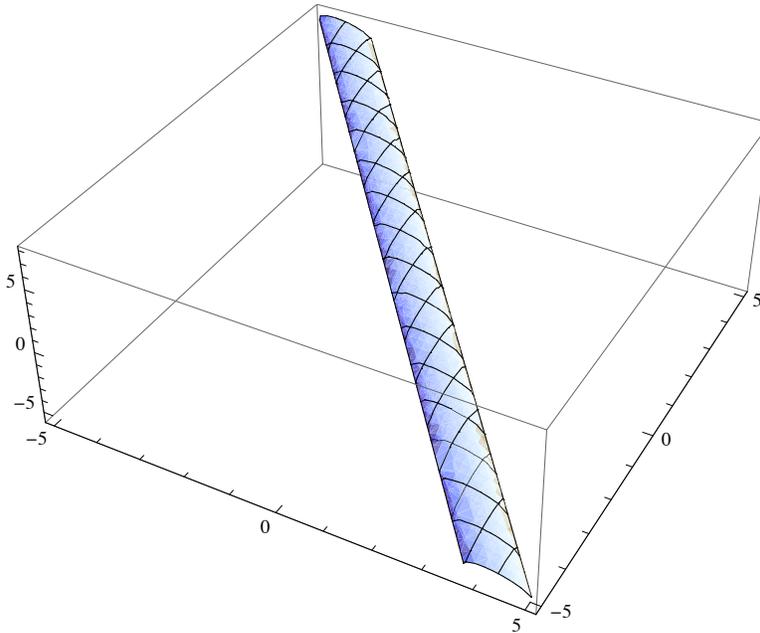
$$\left\{ \left\{ z \rightarrow \frac{1}{226} \left(226 - 140 x + 160 y - \sqrt{213} \sqrt{113 - 256 x^2 - 448 x y - 196 y^2} \right) \right\}, \right. \\ \left. \left\{ z \rightarrow \frac{1}{226} \left(226 - 140 x + 160 y + \sqrt{213} \sqrt{113 - 256 x^2 - 448 x y - 196 y^2} \right) \right\} \right\}$$

Plotting the two branches of the solutions,

$$ss1 = \text{Plot3D}\left[\frac{1}{226} \left(226 - 140x + 160y - \sqrt{213} \sqrt{113 - 256x^2 - 448xy - 196y^2}\right), \{x, -5, 5\}, \{y, -5, 5\}\right]$$

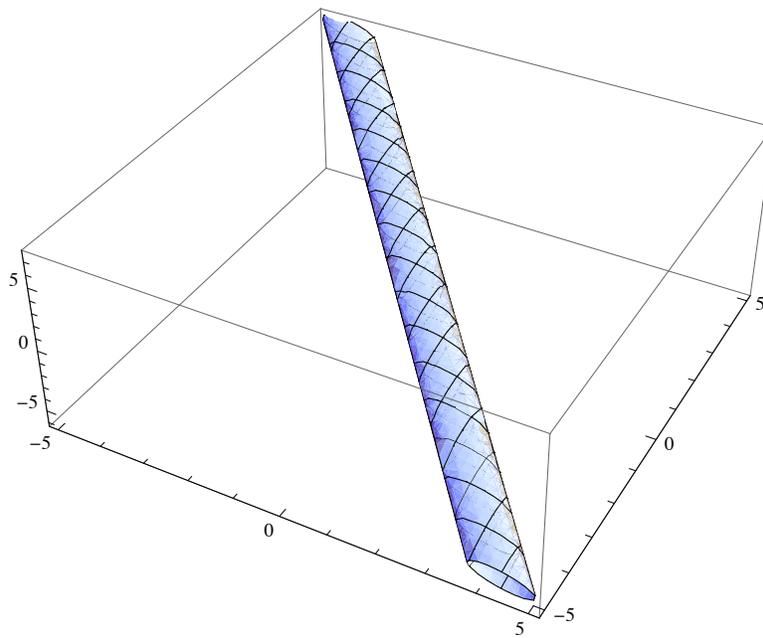


$$ss2 = \text{Plot3D}\left[\frac{1}{226} \left(226 - 140x + 160y + \sqrt{213} \sqrt{113 - 256x^2 - 448xy - 196y^2}\right), \{x, -5, 5\}, \{y, -5, 5\}\right]$$



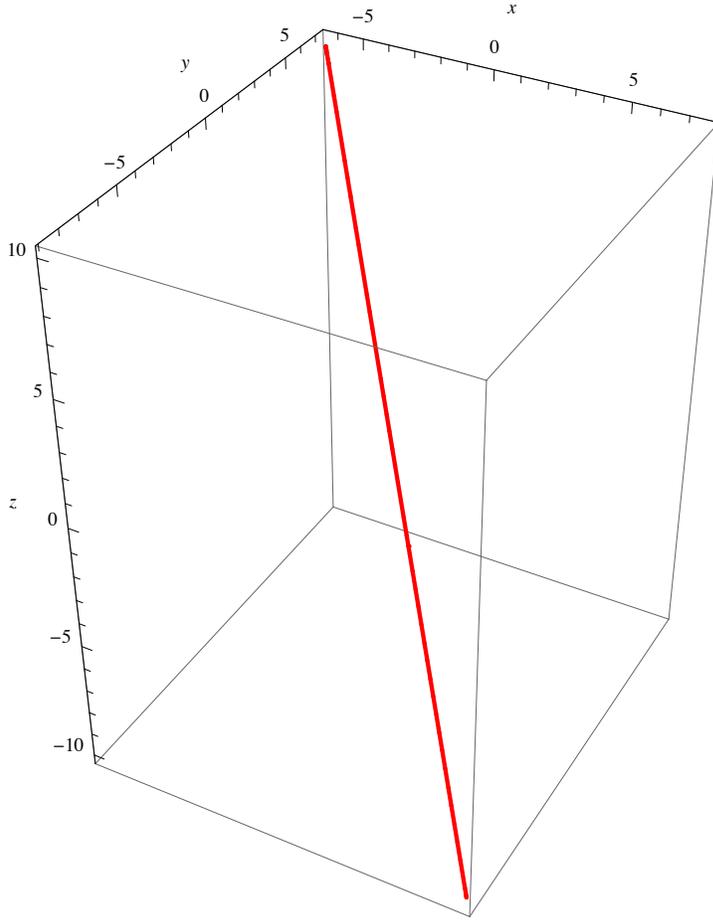
Plot ss1 and ss2 together (they become the entire cylinder),

```
ss = Show[ss1, ss2]
```



Plot the intersection line for $-10 < z < 10$,

```
linePlot =
  ParametricPlot3D[eqline, {z, -10, 10}, PlotStyle -> {Thick, Red}, AxesLabel -> {x, y, z}]
```



Define two points on the line when $z = -3$ and 3 ,

$$\mathbf{point1} = \left\{ -\frac{7}{10} (-1 + (-3)), \frac{4}{5} (-1 + (-3)), -3 \right\}$$

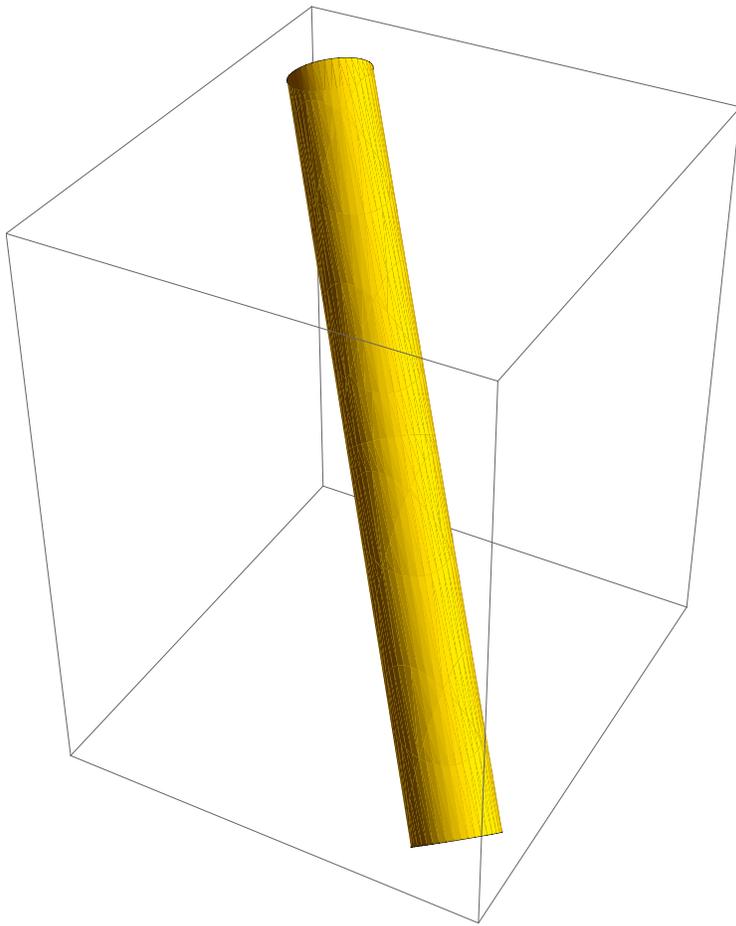
$$\left\{ \frac{14}{5}, -\frac{16}{5}, -3 \right\}$$

$$\mathbf{point2} = \left\{ -\frac{7}{10} (-1 + 3), \frac{4}{5} (-1 + 3), 3 \right\}$$

$$\left\{ -\frac{7}{5}, \frac{8}{5}, 3 \right\}$$

Plot the cylinder between point1 and point2 with the diameter of 1,

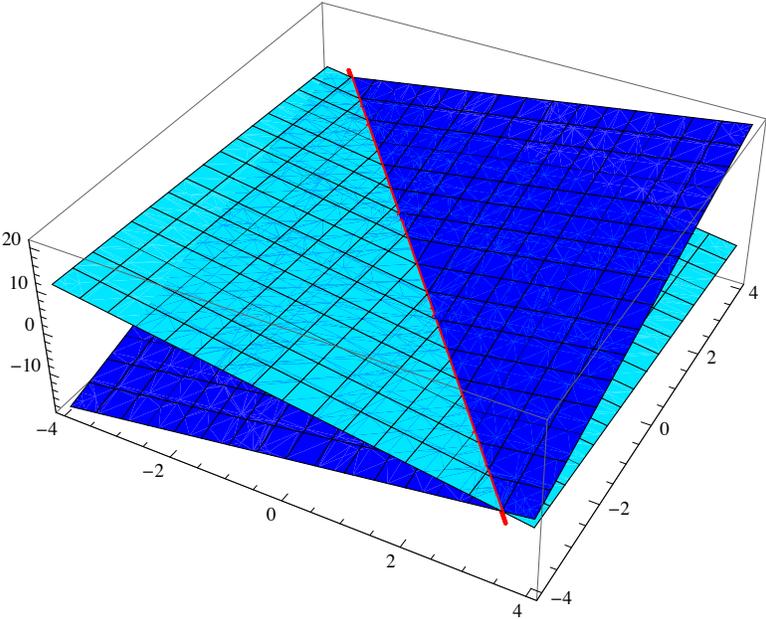
```
cylinderline = Graphics3D[{Yellow, Cylinder[{point1, point2}, 0.5]}]
```



I2-2

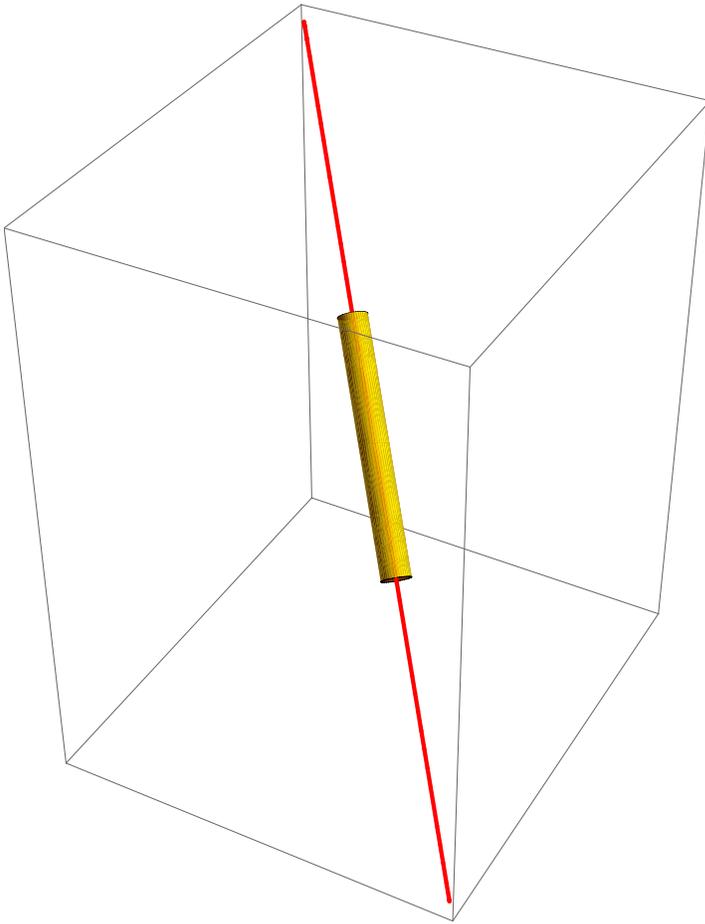
Show the intersection line together with the two planes,

```
Show[planesPlot, linePlot]
```



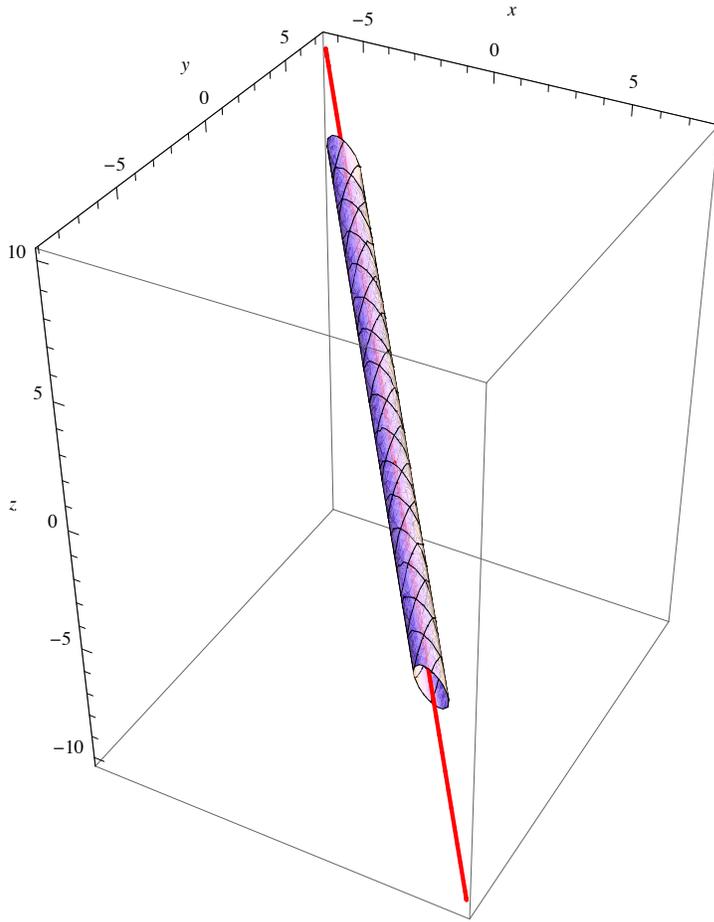
Demonstrate that the cylinder and the intersection line are co-axial,

```
Show[cylinderline, linePlot]
```



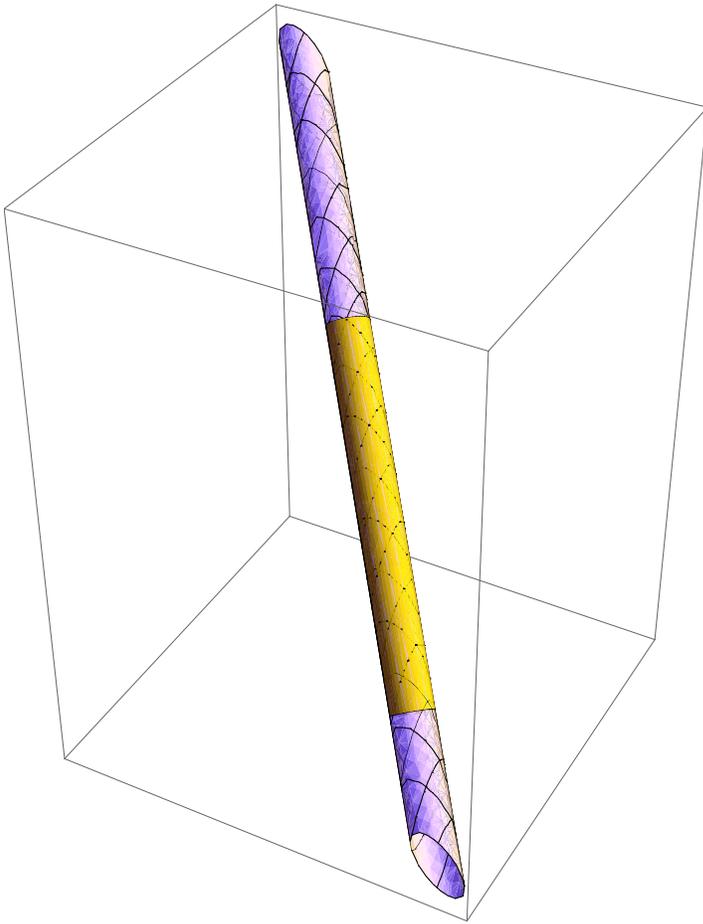
Demonstrate that the solution cylinder **ss** and the intersection line are co-axial,

```
Show[linePlot, ss]
```



Demonstrate that the solution cylinder `ss` and the plotted `cylinderline` are overlapping,

```
Show[cylinderline, ss]
```



I2-3

Assign `aDate` and evaluate,

```
aDate = DateString[]
Fri 24 Sep 2010 15:40:05
```

Check to see what is `aDate` when it is called,

```
Table[Pause[5]; aDate, {5}] // TableForm
Fri 24 Sep 2010 15:40:05
```

Assign `theDate` with the actual time/date whenever it is called,

```
theDate := DateString[]
```

```
Table[Pause[5]; theDate, {5}] // TableForm
```

```
Fri 24 Sep 2010 15:43:06
Fri 24 Sep 2010 15:43:11
Fri 24 Sep 2010 15:43:16
Fri 24 Sep 2010 15:43:21
Fri 24 Sep 2010 15:43:26
```

I2-4

- Consider a small example

```
Table[Fibonacci[i], {i, 1, 20}]
{1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, 233, 377, 610, 987, 1597, 2584, 4181, 6765}

Cases[Table[Fibonacci[i], {i, 1, 20}], a_ /; IntegerQ[Sqrt[a]]]
{1, 1, 144}
```

- Generalize this

```
squaresOfFib[toSize_] := Cases[Table[Fibonacci[i], {i, 1, toSize}], a_ /; IntegerQ[Sqrt[a]]]

squaresOfFib[1000]
{1, 1, 144}
```

- It appears that 144^2 is the only perfect square of the Fibonacci sequence in the first 1000. Computing it for larger lists is expensive with this method. We would wonder if 144 is the last one, and if there is a faster method? This is a problem for another day.

I2-5 (Note: students do not have to do the Viete's formula part, it was poorly worded on the problem set)

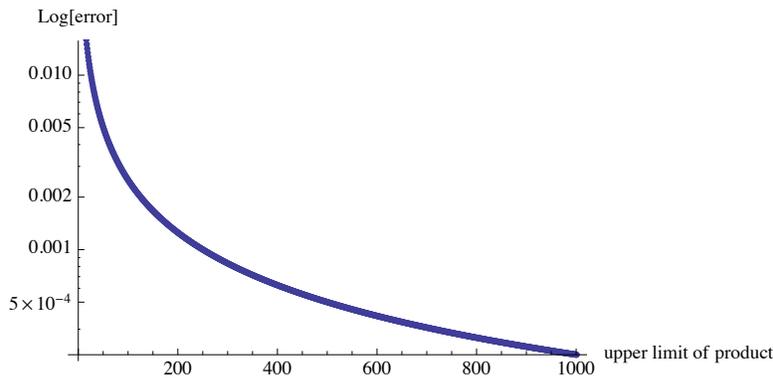
- Wallis's product

```
Product[(2 i / (2 i - 1)) (2 i / (2 i + 1)), {i, 1, Infinity}]

$$\frac{\pi}{2}$$

approxOneWallis[upperlim_] := 2 Product[(2 i / (2 i - 1)) (2 i / (2 i + 1)), {i, 1, upperlim}] / Pi
```

```
convergencePlotWallis = ListLogPlot[Table[1 - approxOneWallis[i], {i, 0, 1000}],
  AxesLabel -> {"upper limit of product", "Log[error]"}]
```



The above gives us a visual representation of the convergence.

- **Viète's formula, let's try to create a function that will give the n-th term of the product sequence of Viète's formula**

```
Clear[vièteNum]; vièteNum[i_] := vièteNum[i] = Sqrt[2 + vièteNum[i - 1]]
```

```
vièteNum[0] = Sqrt[2]
```

$$\sqrt{2}$$

```
Table[vièteNum[i], {i, 1, 5}]
```

$$\left\{ \sqrt{2 + \sqrt{2}}, \sqrt{2 + \sqrt{2 + \sqrt{2}}}, \sqrt{2 + \sqrt{2 + \sqrt{2 + \sqrt{2}}}}, \right.$$

$$\left. \sqrt{2 + \sqrt{2 + \sqrt{2 + \sqrt{2 + \sqrt{2}}}}}, \sqrt{2 + \sqrt{2 + \sqrt{2 + \sqrt{2 + \sqrt{2 + \sqrt{2}}}}}} \right\}$$

These look like the numerators of the form of Viète's formula, let's define the product recursively

```
productViète[i_] := productViète[i] = productViète[i - 1] (vièteNum[i] / 2);
productViète[1] = vièteNum[1] / 2
```

$$\frac{\sqrt{2 + \sqrt{2}}}{2}$$

```
productViète[4]
```

$$\frac{1}{16} \sqrt{\left(2 + \sqrt{2}\right) \left(2 + \sqrt{2 + \sqrt{2}}\right) \left(2 + \sqrt{2 + \sqrt{2 + \sqrt{2}}}\right) \left(2 + \sqrt{2 + \sqrt{2 + \sqrt{2 + \sqrt{2}}}}\right)}$$

Things will be speeded up considerably here if each of the terms doesn't need to be represented as a complicated symbolic expression, let's define a numerical version of the same function

```
Clear[vieteNumNumerical];
vieteNumNumerical[i_] := vieteNumNumerical[i] = Sqrt[(2.0 + vieteNumNumerical[i - 1])];
vieteNumNumerical[0] = Sqrt[2.0]

1.41421

productVieteNumerical[i_] :=
  productViete[i] = productViete[i - 1] (vieteNumNumerical[i] / 2);
productVieteNumerical[1] = vieteNumNumerical[1] / 2

0.92388

productVieteNumerical[4]

0.900678
```

This gives correct numerical values using Viete's formula. Let's explore convergence

```
approxOneViete[upperlim_] := Sqrt[2] Pi productVieteNumerical[upperlim] / 4
```

```
approxOneViete[2]
```

```
1.00645
```

```
Table[1 - approxOneViete[i], {i, 1, 10}]
```

```
{-0.0261722, -0.00645454, -0.00160819, -0.000401708, -0.000100406,
-0.0000251001, -6.27495 × 10-6, -1.56873 × 10-6, -3.92183 × 10-7, -9.80457 × 10-8}
```

```
convergencePlotVietes = ListLogPlot[Table[Abs[1 - approxOneViete[i]], {i, 1, 30}],
  AxesLabel → {"upper limit of product", "Log[error]"}]
```

