Example of LaTeX on Cocalc

July 3, 2018

This is an example $2 + 2 = 4$. If you raise the current year mod 100 (which equals 18) to the power of the current day (3), you get 5832. Also, 2018 modulo 42 is 2.

Code block which uses a variable `s` to store the solutions:

```
1+1
var('a,b,c')
eqn = [a+b*c==1, b-a*c==0, a+b==5]
s = solve(eqn, a,b,c)
```

Solutions of eqn $= [bc + a = 1, -ac + b = 0, a + b = 5]$:

$$\left[a = -\frac{1}{4}i\sqrt{79} + \frac{11}{4}, b = \frac{1}{4}i\sqrt{79} + \frac{9}{4}, c = \frac{1}{10}i\sqrt{79} + \frac{1}{10}\right]$$

$$\left[a = \frac{1}{4}i\sqrt{79} + \frac{11}{4}, b = -\frac{1}{4}i\sqrt{79} + \frac{9}{4}, c = -\frac{1}{10}i\sqrt{79} + \frac{1}{10}\right]$$

Now we evaluate the following block:

```
E = EllipticCurve("37a")
```

You can't do assignment inside `\sage` macros, since Sage doesn't know how to typeset the output of such a thing. So you have to use a code block. The elliptic curve E given by $y^2 + y = x^3 - x$ has discriminant 37.

You can do anything in a code block that you can do in Sage and/or Python. Here we save an elliptic curve into a file.

```
try:
    E = load('E2')
except IOError:
    E = EllipticCurve([1,2,3,4,5])
    E.anlist(100000)
    E.save('E2')
```

The 9999th Fourier coefficient of $y^2 + xy + 3y = x^3 + 2x^2 + 4x + 5$ is -27.

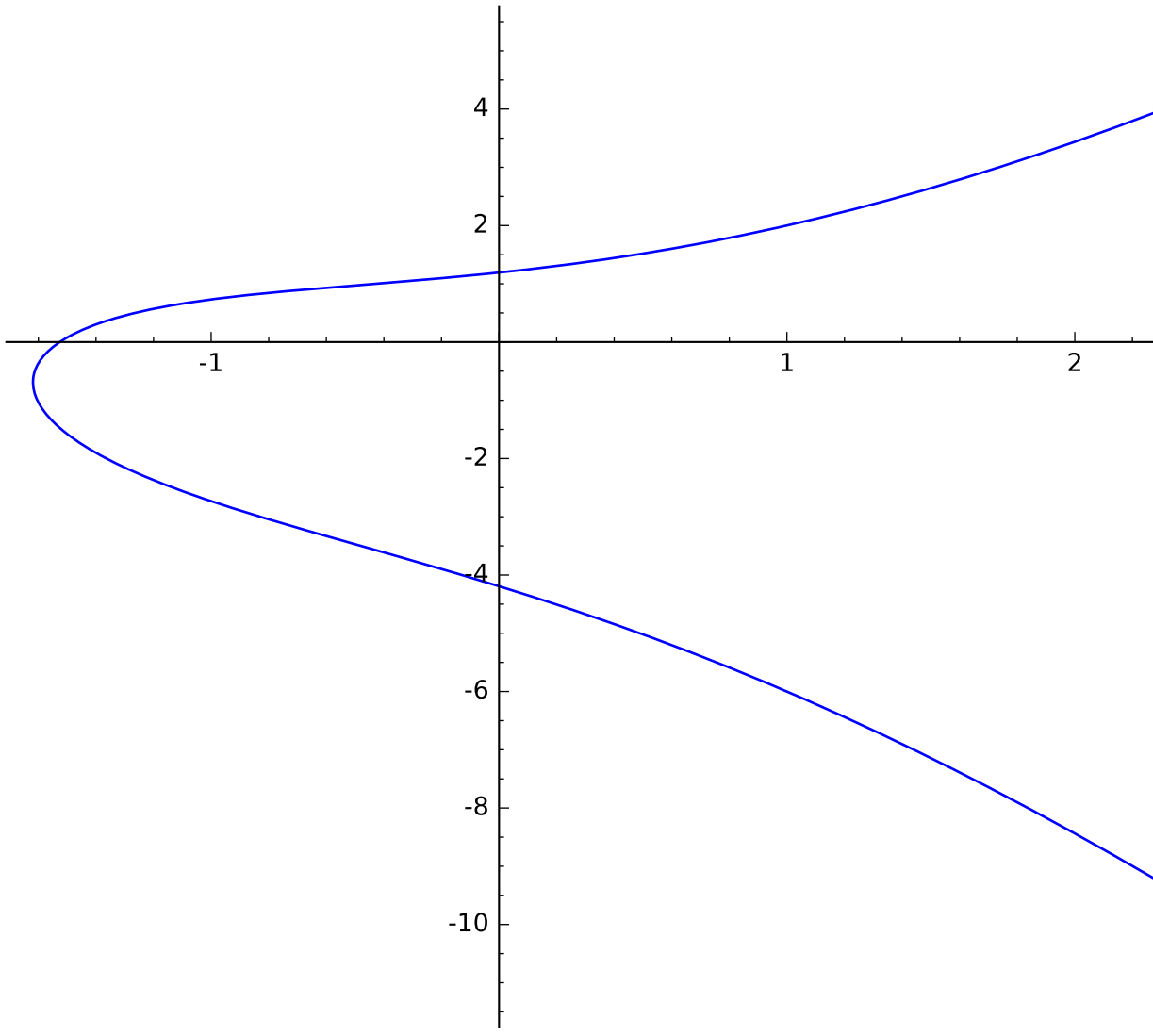
The following code block doesn't appear in the typeset file... but we can refer to whatever we did in that code block: $e = 7$.

```
var('x')
f(x) = log(sin(x)/x)
```

The Taylor Series of f begins: $x \mapsto -\frac{1}{467775}\,x^{10} - \frac{1}{37800}\,x^{8} - \frac{1}{2835}\,x^{6} - \frac{1}{180}\,x^{4} - \frac{1}{6}\,x^{2}$.

1 Plotting

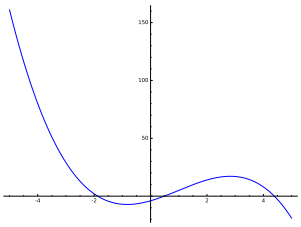
Here's a plot of the elliptic curve E.



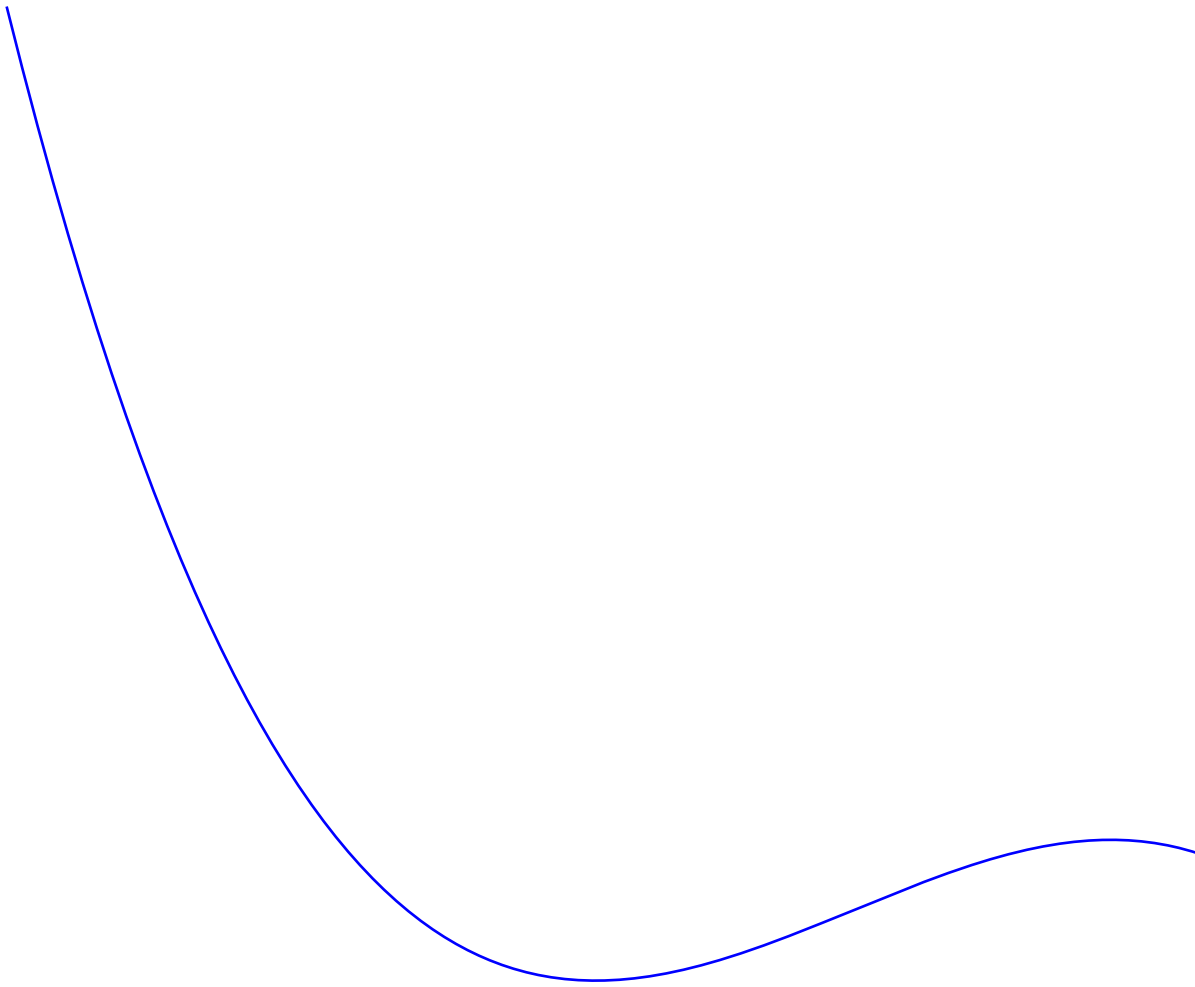
You can use variables to hold plot objects and do stuff with them.

```
p = plot(f, x, -5, 5)
```

Here's a small plot of f from -5 to 5, which I've centered:



On second thought, use the default size of 3/4 the `\textwidth` and don't use axes:

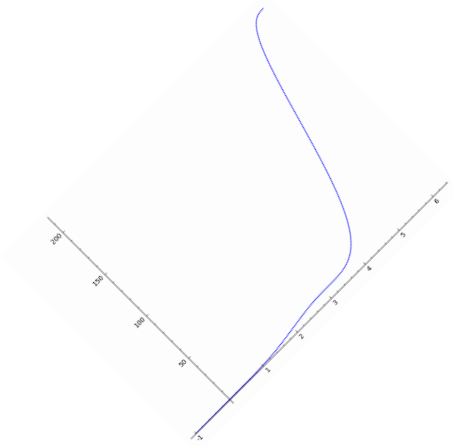


Remember, you're using Sage, and can therefore call upon any of the software packages Sage is built out of.

```
f = maxima('sin(x)^2*exp(x)')
g = f.integrate('x')
```

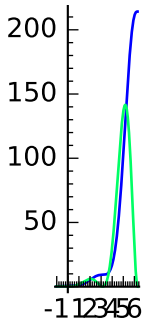
Plot $g(x)$, but don't typeset it.

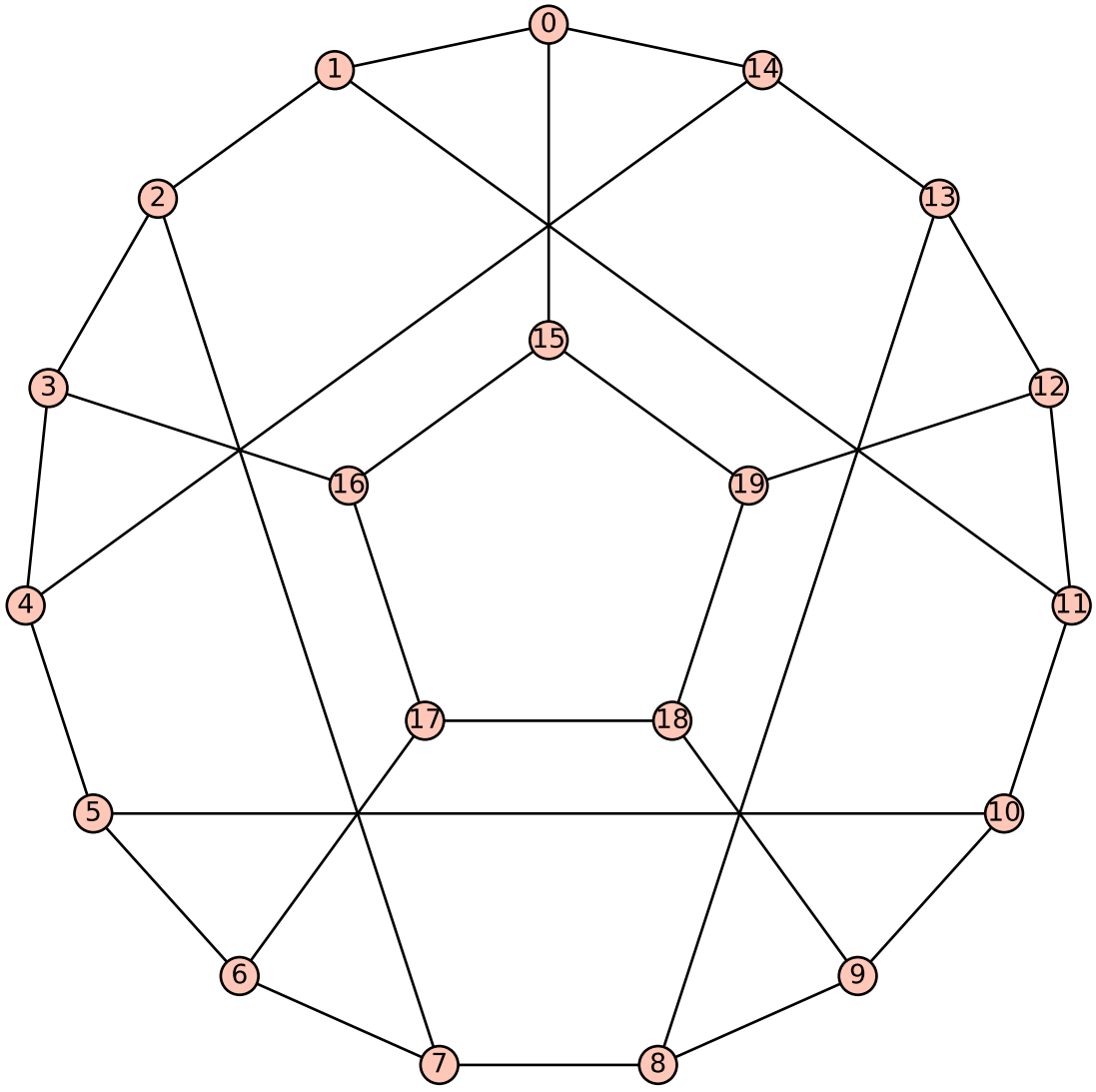
You can specify a file format and options for `includegraphics`. The default is for EPS and PDF files, which are the best choice in almost all situations. (Although see the section on 3D plotting.)



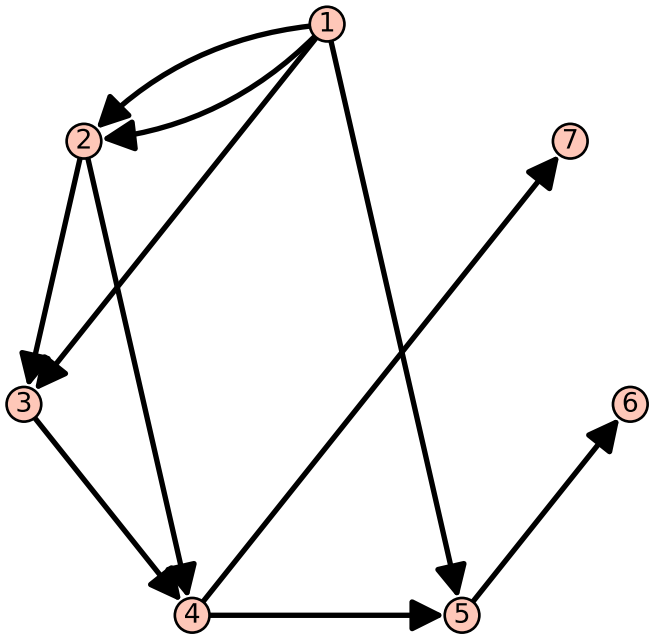
If you use regular `latex` to make a DVI file, you'll see a box, because DVI files can't include PNG files. If you use `pdflatex` that will work. See the documentation for details.

When using `\sageplot`, you can pass in just about anything that Sage can call `.save()` on to produce a graphics file:






```
G4 = DiGraph({1:[2,2,3,5], 2:[3,4], 3:[4], 4:[5,7], 5:[6]},\
             multiedges=True)
G4plot = G4.plot(layout='circular')
```

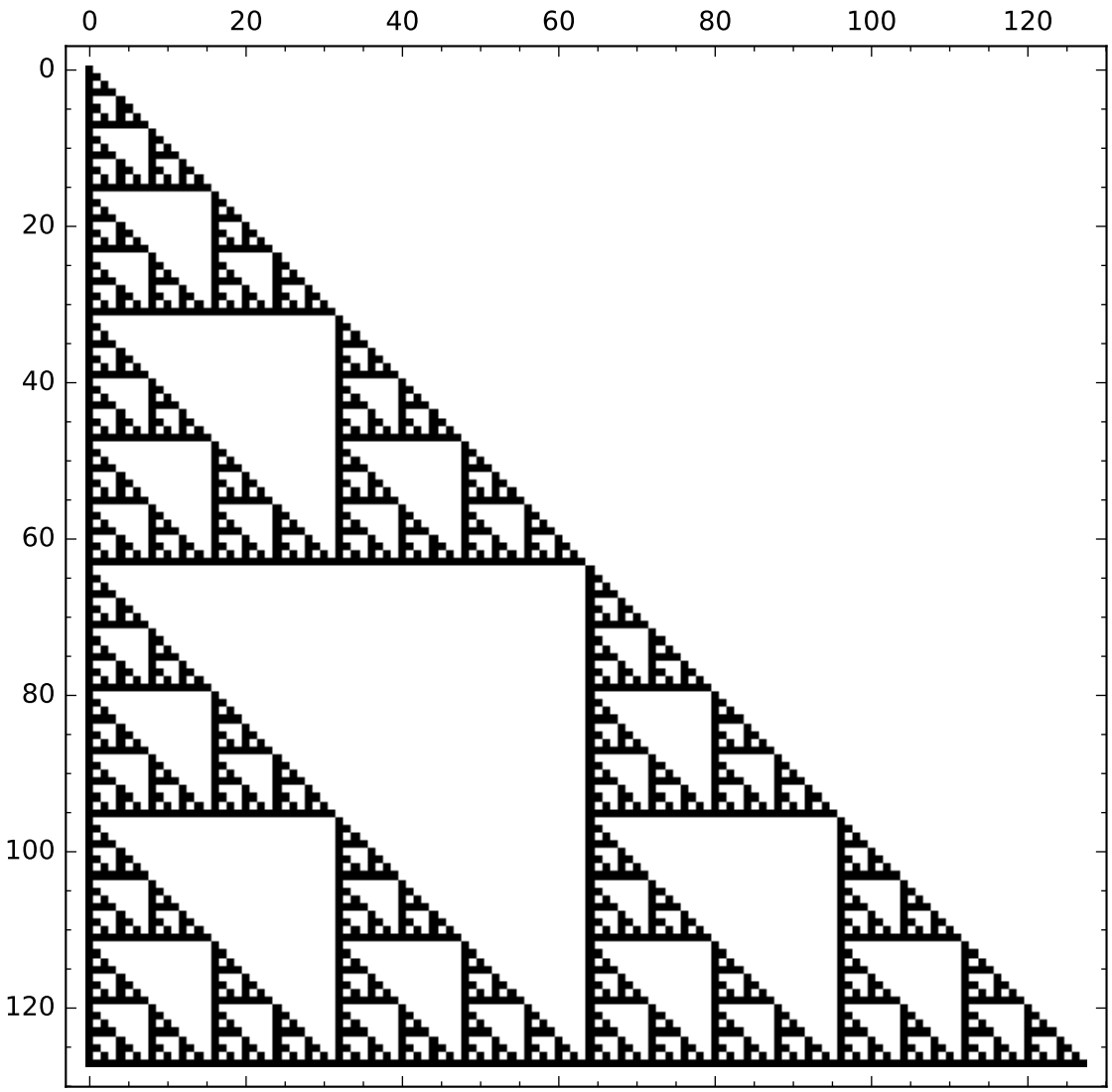


Indentation and so on works fine.

```
s     = 7
s2    = 2^s
P.<x> = GF(2)[]
M     = matrix(parent(x),s2)
for i in range(s2):
   p  = (1+x)^i
   pc = p.coeffs()
   a  = pc.count(1)
   for j in range(a):
       idx        = pc.index(1)
       M[i,idx+j] = pc.pop(idx)

matrixprogram = matrix_plot(M,cmap='Greys')
```

And here's the picture:


0
20
40
60
80
100
120
0
20
40
60
80
100
120