Lab: 8
Doing Algebra with Sage

Most of the computations we did in the previous section can be done on a regular calculator, but where a CAS like Sage really shines is when working with algebraic expressions including variables. When working with Sage, we will need to declare symbolic variables before we use them.

```
x = var('x')
```

This tells Sage that we will be using the symbol/letter "x" as an unbound variable in an expression. We wish to treat it as symbol that can represent any possible value and not as a specific value that is fixed. In other words, x is a mathematical variable, not a computer variable.

```
x = var('x')
3*x+7*x+5
```

This represents the expression $3x + 7x + 5$. Note that the implied multiplication between 3 and x needs to be specified when typing into Sage.

Inputting

```
var('x')
3x
```

will just cause a NameError.

Sage can be helpful when working with algebraic expressions, as it can do things like expand, factor, or simplify expressions.

If we want to factor an expression like $x^2 + 4x + 3$, we can use the `factor` command.

```
x=var('x')
factor(x^2+4*x+3)
```

We can also expand out an expression using the `expand` command.

```
x=var('x')
expand( (x+1)*(x+3) )
```

It is also often useful to define a function, such as $f(x)$, much like we do in a regular math class.

```
x=var('x')
f(x)=x^2+4*x+3
```

We can then evaluate $f(x)$ at different values of x using the standard notation. For example, to evaluate at $x = 1$, we can then enter

```
f(1)
```

which will give us the output of

```
8
```

which is the same as the value of $(1)^2 + 4(1) + 3$, i.e. the value of substituting $x = 1$ into $f(x) = x^2 + 4x + 3$.

We can also easily apply the **expand** and **factor** commands to a function we have already defined.

```
f.factor()
```

will give the output

```
x |--> (x+3)*(x+1)
```

which is the same factorization as when we typed

```
factor(x^2+4*x+3)
```

Similarly,

```
f.expand()
```

can be used to **expand** the function $f(x)$. In addition, a function has the **full_simplify** option.

```
var('x')
f(x)=(x^2+4*x+3)/x+(x+3)*(x-1)/(x+1)
f.full_simplify()
```

Will simplify the very complicated expression of

$$\frac{x^2 + 4x + 3}{x} + \frac{(x+3)(x-1)}{x+1}$$

into a single rational function.

Finally, sometimes Sage will give us an exact expression for something for which we would like a decimal approximation. For example,

```
var('x')
f(x)=(x^2+4*x+3)/x+(x+3)*(x-1)/(x+1)
f(2)
```

gives the output

```
55/6
```

Because Sage is mathematical software, and mathematicians usually want exact answers, that is what it will return, when possible. In order to force it to give a decimal expression, we can use the `n()` command.

```
n(55/6)
```

gives the decimal approximation of

```
9.16666666666667
```