```
>
> dumb:=proc(x)
    local y;
    y:=x^1380;
    y:=modp(y^3, 12);
  end:
> dumb(17);
```

$$1 \quad (1)$$

```
> dumb:=proc(x)
    local y;
    y:=x^13;  print("y is",y,"x is",x);
    y:=modp(y^3, 12);
  end:
> dumb(18);
```

"y is", 20822964865671168, "x is", 18

$$0 \quad (2)$$

```
> debug(dumb);
```

$$dumb \quad (3)$$

```
> dumb(5);
{--> enter dumb, args = 5
```

$$y := 1220703125$$

"y is", 1220703125, "x is", 5

$$y := 5$$

```
<-- exit dumb (now at top level) = 5}
```

$$5 \quad (4)$$

```
> ?debugger
```

Fermat's little theorem: if p is prime, $a^p = a \ \mathbf{mod}\ p$
(or, if gcd(a,p)=1, then $a^{p-1} = 1 \ \mathbf{mod}\ p$)

```
> p:=7;
```

$$p := 7 \quad (5)$$

```
> A:=4;
```

$$A := 4 \quad (6)$$

```
> seq( [i,A^i mod p], i=1..15);
```

[1, 4], [2, 2], [3, 1], [4, 4], [5, 2], [6, 1], [7, 4], [8, 2], [9, 1], [10, 4], [11, 2], [12, 1], [13, 4], [14, 2], [15, 1] (7)

```
> A:=3;seq( [i,A^i mod p], i=1..15);
```

$$A := 3$$

[1, 3], [2, 2], [3, 6], [4, 4], [5, 5], [6, 1], [7, 3], [8, 2], [9, 6], [10, 4], [11, 5], [12, 1], [13, 3], [14, 2], [15, 6] (8)

what is
$a \cdot a^2 \cdot a^3 \cdot a^4 \cdot a^5 \cdot a^6 \ \mathbf{mod}\ p$

it is 3*2*6*4*2*1 == 1*2*3*4*5*6

what about mod n where n=pq?

Euler's Theorem

n=pq, where p, q are prime.

Then $a^{(p-1)(q-1)} = 1 \mod n$

if gcd(n,a)=1

RSA:

Choose two big primes p & q

Let f = (p-1)(q-1), n=p*q

Choose e [my exponent] so that gcd(e,f) = 1

Calculate d = 1/e mod f

publish (e, n) --- public key

secret (d) - --- private key

To encode a message m, calculate $m^e \mod n = x$

To decode x, calculate $x^d \mod n = m$

```
> p:=nextprime(20); q:=nextprime(30);
```

$$p := 23$$

$$q := 31 \quad (9)$$

```
> f:=(p-1)*(q-1);  n:=p*q;
```

$$f := 660$$

$$n := 713 \quad (10)$$

```
> e:=47;
```

$$e := 47 \quad (11)$$

```
> gcd(47,f);
```

$$1 \quad (12)$$

```
> d:=1/e mod f;
```

$$d := 323 \quad (13)$$

```
> m:=18;
```

$$m := 18 \quad (14)$$

```
> m^e mod n;
```

$$634 \quad (15)$$

```
> 634^d mod n;
```

$$18 \quad (16)$$

```
> m^123125312541253789064361463552530123412735627561209812571230957823908239085620139856239085620398561283905 mod n
```

Warning, inserted missing semicolon at end of statement
Error, numeric exception: overflow

```
> m&^123125312541253789064361463552530123412735627561209812571230957
  823908239085620139856239085620398561283905 mod n;
```

$$377 \quad (17)$$

```
> int(e^sin(x),x=1..10);
```

$$\int_{1}^{10} 47^{\sin(x)}\, dx \quad (18)$$

```
> Int(e^sin(x),x=1..10);
```

$$\int_{1}^{10} 47^{\sin(x)}\, dx \quad (19)$$

```
> int(x^10,x=1..10);
```

$$\frac{99999999999}{11} \quad (20)$$

```
> Int(x^10,x=1..10);
```

$$\int_{1}^{10} x^{10}\, dx \quad (21)$$

Why does RSA work? (assuming you know Euler's theorem)

for notation, use [x] to mean x mod n

$$\left[[m^e]^d \right] = [m^{ed}] = [m^{1+kf}] = [m]^1 [m^{kf}] = m \left[[m^f]^k \right] = m[1]^k = m$$