Awesome
Operators

In Perl 6 we have a few new operators...

Junctions

```
if $status eq 'error' || $status eq 'warning' {
    ...
}
```

Junctions

```
if $status eq 'error' || $status eq 'warning' {
    ...
}
```

```
if $status eq 'error' | 'warning' {
    ...
}
```

Junctions

```
while $value < $limit1 && $value < $limit2 {
    ...
}
```

Junctions

```
while $value < $limit1 && $value < $limit2 {
    ...
}




while $value < $limit1 & $limit2 {
    ...
}
```

Sequences

```
say 1, 2, 4 ... 1024;
```

1 2 4 8 16 32 64 128 256 512 1024

Sequences

```
my @fib = 1, 1, *+* ... *;
say @fib[0..9]
```

1 1 2 3 5 8 13 21 34 55 89

^ (zero up to...)

```
my @fib = 1, 1, *+* ... *;
say @fib[^10]
```

1 1 2 3 5 8 13 21 34 55 89

Awesome
Meta-operators

Higher order
operators

Operators that operate on operators

Reduction

Puts an operator between every element in a list

```
say [*] 1..10
```

3628800

Reduction

```
my @sorted = 1,4,7,9,11;
my @unsorted = 3, 1, 9, 25;

say [<] @sorted;
say [<] @unsorted;

Bool::True
Bool::False
```

Zip

Take elements from two or more lists and combine them with some operator

```
say 1 .. 6 Z~ 'A'..'F'
```

1A 2B 3C 4D 5E 6F

Cross

All permutations of two or more lists, combined with some operator

```
say 1 .. 3 X~ 'A'..'F'
```

1A 1B 1C 1D 1E 1F 2A 2B 2C 2D 2E 2F
3A 3B 3C 3D 3E 3F

Your
Awesome
Operators

What if Perl 6 built in operators are **not enough**?

You can
add your own!

Factorial

Add a ! operator to do factorial

```
sub postfix:<!>($n) { [*] 1..$n }
say 10!
```
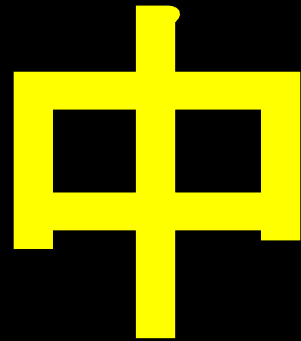
3628800

And you have
all of unicode!

Insert In Middle

Operator to add an element to the middle of an array

Insert In Middle

Operator to add an element to the middle of an array

中

Insert In Middle

```
sub infix:<中>(@array, $ins) {
    @array.splice(+@array / 2, 0, $ins);
    return @array;
}

my @a = 1,2,4,5;
say @a 中 3;
```

1 2 3 4 5

Thanks!

☺