5 Math

Key terms: `**` `//` `%` `math` `log` `random`

Reading: WikiBooks Python Programming's "Basic Math"[10]

Exercise: Write a program that prints out a number so great that it is at least ten lines long in your terminal.

Besides the classic four arithmetic operators (`+`, `-`, `*`, `/` or `//`), several more useful infix operators are available. `**` raises a base number to a power — and since floating point exponents work fine, roots and reciprocals can be found with the same operator (e.g., `2**0.5`). `round()` is a builtin function.

```
# Print a high power of two.
print ( 2 ** 10 )

# Print a fractional power of two and the same, rounded.
print ( 2 ** 0.1, round ( 2 ** 0.1 ) )
```

Python has two division operators which return different types when operating on integers. `/`, a single slash, always returns a `float`. `//`, a double slash, always returns an `int`, perhaps leaving some remainder. Correspondingly, `%`, modulo, indicates that remainder. This arrangement repairs the longstanding confusion over the C-style division operator whose return type depends on the type of operands — a behavior not universally desired. We now enjoy more flexibility.

```
# Share out balls equally between jugglers.
# Integer division is used because balls stay whole.

jugglers = 6
balls = 100

balls_per_juggler = balls // jugglers
balls_left_over = balls % jugglers
```

The `math` module offers common operations among which are those for logarithms (`log`, `log2`, `log10`) and trigonometry (`sin`, `cos`, `tan`, `asin`, `acos`, `atan`). The last one, `atan`, is particularly useful — it determines the angle under a hypotenuse extending from the origin to any X, Y coordinates. The transcendental constants `e` and `pi` are defined too.

```
# Verify that logarithms and exponentiation work properly.

import math
print ( math.pi == math.log10 ( 10 ** math.pi ) )
```

Pseudorandomness is available in the `random` package. This is a feature you can expect to use in the future as it has many applications. The principal unit of randomness, acquired from the `random()` method, is a positive `float` under 1. You can also call on `randint()` for random integers, `choice()` for a single choice from a collection, or `sample()` to take a random sample.

```
import random

# Model the roll of an 18 sided die.

die_roll = random.randint(1,18)
print(die_roll)

# Model a random point on the unit line.

point = random.random()
print(point)
```

Next, we will address the basic container types.

[10] `https://en.wikibooks.org/wiki/Python_Programming/Basic_Math`