Diving In

Just as regular expressions put strings on steroids, the `itertools` module puts iterators on steroids. But first, I want to show you a classic puzzle.

```
HAWAII + IDAHO + IOWA + OHIO == STATES
510199 + 98153 + 9301 + 3593 == 621246

H = 5
A = 1
W = 0
I = 9
D = 8
O = 3
S = 6
T = 2
E = 4
```

Puzzles like this are called *cryptarithms* or *alphametics*. The letters spell out actual words, but if you replace each letter with a digit from `0–9`, it also “spells” an arithmetic equation. The trick is to figure out which letter maps to each digit. All the occurrences of each letter must map to the same digit, no digit can be repeated, and no “word” can start with the digit 0.

The most well-known alphametic puzzle is *SEND + MORE = MONEY*.

In this chapter, we’ll dive into an incredible Python program originally written by Raymond Hettinger. This program solves alphametic puzzles *in just 14 lines of code*.

```
import re
import itertools

def solve(puzzle):
    words = re.findall('[A-Z]+', puzzle.upper())
    unique_characters = set(''.join(words))
    assert len(unique_characters) <= 10, 'Too many letters'
    first_letters = {word[0] for word in words}
    n = len(first_letters)

    sorted_characters = ''.join(first_letters) + \
        ''.join(unique_characters - first_letters)
    characters = tuple(ord(c) for c in sorted_characters)
    digits = tuple(ord(c) for c in '0123456789')
    zero = digits[0]
    for guess in itertools.permutations(digits, len(characters)):
        if zero not in guess[:n]:
            equation = puzzle.translate(dict(zip(characters, guess)))
            if eval(equation):
                return equation

if __name__ == '__main__':
    import sys
    for puzzle in sys.argv[1:]:
        print(puzzle)
        solution = solve(puzzle)
        if solution:
            print(solution)
```

You can run the program from the command line. On Linux, it would look like this. (These may take some time, depending on the speed of your computer, and there is no progress bar. Just be patient!)

```
you@localhost:~/diveintopython3/examples$ python3 alphametics.py "HAWAII + IDAHO + IOWA + OHI
HAWAII + IDAHO + IOWA + OHIO = STATES
510199 + 98153 + 9301 + 3593 == 621246

you@localhost:~/diveintopython3/examples$ python3 alphametics.py "I + LOVE + YOU == DORA"
I + LOVE + YOU == DORA
1 + 2784 + 975 == 3760

you@localhost:~/diveintopython3/examples$ python3 alphametics.py "SEND + MORE == MONEY"
SEND + MORE == MONEY
9567 + 1085 == 10652
```