Base Sequences and Digit Sequences

Consider the number

$$\frac{25}{8} = 3.125.$$

We can interpret this number as

$$\frac{25}{8} = 3 + \frac{1}{10} + \frac{2}{100} + \frac{5}{1000}$$

or as

$$\frac{25}{8} = 3 + \frac{1}{10}(1 + \frac{1}{10}(2 + \frac{1}{10}(5))).$$

Another representation for $\frac{25}{8}$ then is with a *base sequences* $b = (\frac{1}{10}, \frac{1}{,}10\frac{1}{10})$ and a *digit sequences* $a = (3; 1, 2, 5)$. We can extend this to infinite base and digit sequences as well. For example, consider the transcendental number $e = 2.718281828459045...$ which we can interpret as

$$e = 2 + \frac{7}{10} + \frac{1}{100} + \frac{8}{1000} \cdots$$

or as

$$e = 2 + \frac{1}{10}\left(7 + \frac{1}{10}\left(1 + \frac{1}{10}\left(8 + \cdots\right)\right)\right).$$

The *base sequence* for the above representation is $b = (\frac{1}{10}, \frac{1}{10}, \ldots)$ and the digits are $(2; 7, 1, 8, \ldots)$.

Mixed Bases

In the above two examples, the base or *radix* at each index is identical. But one could also consider base sequences that are *mixed*. For example, suppose $b = (\frac{1}{2}, \frac{1}{3}, \frac{1}{4}, \ldots)$. Now consider the digit sequence $(2; 1, 1, 1 \ldots)$. This corresponds to

$$2 + \frac{1}{2}\left(1 + \frac{1}{3}\left(1 + \frac{1}{4}\left(1 + \cdots\right)\right)\right) = \sum_{i=0}^{\infty} \frac{1}{i!} = e.$$

We've traded a constant base representation with a periodic one, but it also means the digits are constant!

Definition 1 (regular)**.** *If b is some base sequence and $(a_0; a_1, a_2, a_3, \ldots)$ are the digits such that $0 \leq a_i \leq i$ for all $i \geq 1$ then the representation is called* regular.

Enumerating Digits

Suppose $x = \frac{25}{8} = 3.125$ and we wanted to enumerate the digits of x. One way would be for us to repeatedly take the integer component (3) and then create a new number by considering only the fractional component (0.125) multiplied by 10 (which yields 1.25). No we can repeat the process. Now consider x in its base sequence format. We can drop by integer part of the number be setting $a_0 = 0$. Notice that

$$10 \times 0.125 = (0; 10 \times 1, 10 \times 2, 10 \times 5) = (0; 10, 20, 50)$$

and b remains the same. Notice this expression corresponds to:

$$0 + \frac{1}{10}(10 + \frac{1}{10}(20 + \frac{1}{10}(50))).$$

which we can simplify to:

$$\begin{aligned}
0 + \frac{1}{10}(10 + \frac{1}{10}(20 + \frac{1}{10}(50))) &= 0 + \frac{1}{10}(10 + \frac{1}{10}(25 + \frac{1}{10}(0))) \\
&= 0 + \frac{1}{10}(12 + \frac{1}{10}(5 + \frac{1}{10}(0))) \\
&= 1 + \frac{1}{10}(2 + \frac{1}{10}(5 + \frac{1}{10}(0))).
\end{aligned}$$

This corresponds to a digit sequence of $(1; 2, 5, 0)$.

A spigot for e

There is nothing special about the multiplication by 10 and the simplification routine we developed in the previous section. We can easily apply it to any mixed base, where our simplification always factors out $1/b_i$ from the digit at a_i. This means, that if we want to enumerate, say, the first n digits of an irrational number, we start with a digit sequence of length n and repeat the algorithm[1].

The big idea is that if we want to know the first n digits of e, then we use mixed-radix base $b = (\frac{1}{2}, \frac{1}{3}, \frac{1}{4}, \dots, \frac{1}{n})$ with digits $(2; \overbrace{1, 1, 1, .., 1}^{n})$, take 2 as our first digit, multiply the fractional part by 10, make the expression regular to recover the new integer digit, and repeat. We need at least $n + 3$ fractional digits to get the first n digits of a number accurately.

Algorithm

The following algorithmic pseudocode comes from [**?**]

Initialize Let the first digit be 2 and let $A = (1, 1, ..., 1)$ be an array of length $n + 3$ (0-indexed).

Loop Repeat $n - 1$ times:

1. Multiple each value in A by 10.
2. From the right, reduce the i^{th} entry of A modulo $i + 2$, carrying the quotient one place left.
3. The final quotient is the next digit of e.

Example

Suppose we want the first 3 digits of e. This require 6 fractional digits. Here is how the algorithm works:

	Base 10	Integer	$1/2$	$1/3$	$1/4$	$1/5$	$1/6$	$1/7$
	2.718...	2	1	1	1	1	1	1
$10\times$	7.18..		10	10	10	10	10	10
carries			4	3	2	1	1	0
	7.18..	7	0	1	0	1	5	3
$10\times$	1.8..		0	10	0	10	50	30
carries			3	0	3	9	4	0
	1.8	1	1	1	3	4	0	2

[1] To be fair, because of rounding errors, we almost surely need more than n digits to correctly generate the first n digits. For e, we need $n + 3$ digits