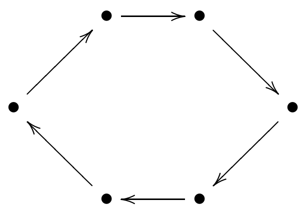
Math 260: Being and octopus

Miguel Carrión Álvarez

1. *Decomposing an octopus.*

An octopus consists of a cyclically ordered set of n "legs" of length at least 1, each linearly ordered. An example where all legs are of length 1 is



Accordingly, let F be the structure type "being a cyclically ordered finite set", and let G be the structure type "being a linearly ordered non-empty set". Then, the structure type "being an octopus" is $\mathrm{Oct} \cong F \circ G$.

2. *Generating functions.*

Observe that $G + 1$ is isomorphic to the structure type L "being a linearly ordered finite set", which has generating function

$$|L|(z) = (1 - z)^{-1}.$$

This means $|G|(z) + 1 = (1 - z)^{-1}$, so

$$|G|(z) = \frac{z}{1 - z}.$$

As we also know, the structure type F, "being a cyclically ordered finite set", has generating function

$$|F|(z) = -\log(1 - z).$$

It follows that

$$|\mathrm{Oct}|(z) = -\log\Big(1 - \frac{z}{1 - z}\Big) = \log \frac{1 - z}{1 - 2z}.$$

3. *Counting octopi.*

Observe that

$$\log \frac{1 - z}{1 - 2z} = \log(1 - z) - \log(1 - 2z) = -\sum_{n \geq 1} \frac{z^n}{n} + \sum_{n \geq 1} \frac{(2z)^n}{n} = \sum_{n \geq 1} \frac{2^n - 1}{n} z^n.$$

It follows that the number of ways to make an n-element set into an octopus is

$$(n - 1)!(2^n - 1) \qquad \text{for} \quad n > 0.$$