Asymptotics

Definition:

- Asymptotic Runtime: bound of execution time of an algorithm when the size of the problem goes to infinity
  - For all cases (both best case and worst case), we are assuming the input size is **infinitely large**.
  - When looking at best-case runtime and worst-case runtime, we want to think about how values in this infinitely large sized input may create different runtime.

2 Types of Asymptotic Questions:

- Time Complexity: How long does it take to run a program if we give it a certain input?
- Space Complexity: How much space does this program take when we run it on our computers?

3 Types of Bounds:

1. Omega Bound (Ω)
   a. Lower bound
   b. Describes the **fastest** a program can run as the size of the input approaches infinity
2. Big O Bound (O)
   a. Upper bound
   b. Describes the **slowest** a program can run as the size of the input approaches infinity
3. Theta Bound (Θ)
   a. Tight bound
   b. Describes how a function behaves on a given set of inputs
   c. If we are describing how a function behaved on just 1 input, we can always use Θ bound to denote it because the outcome of running a function on a specific input is deterministic.
   d. If we are analyzing how the function behaves on various inputs and some inputs performed better than others, then we won't be able to give an overall bound (aka. a Θ bound) for the function. → would have to use omega and big O bounds instead
   e. However, we can use Θ bound when the **upper (big O)** and **lower (omega)** bounds are the **same**!

Rules in Analyzing Asymptotics

- Ignore lower degree terms
- Ignore constant scaling factors
- For exponential runtime, the base matters.
- For polynomial runtime, the exponent matters.
- For logarithmic runtime, the base does not matter.

Useful Formula

- Arithmetic Series
  - $1 + 2 + 3 + \cdots + N = \Theta(N^2)$
- Geometric Series
  - In increasing order, proportional to the last term.
  - $1 + 2 + 4 + 8 + \cdots + N = \theta(N)$
  - $1 + 2 + 4 + 8 + \cdots + 2^N = \theta(2^N)$
- Log Rules
  - Power rule
  - Product rule
  - Quotient rule
  - Change of base rule

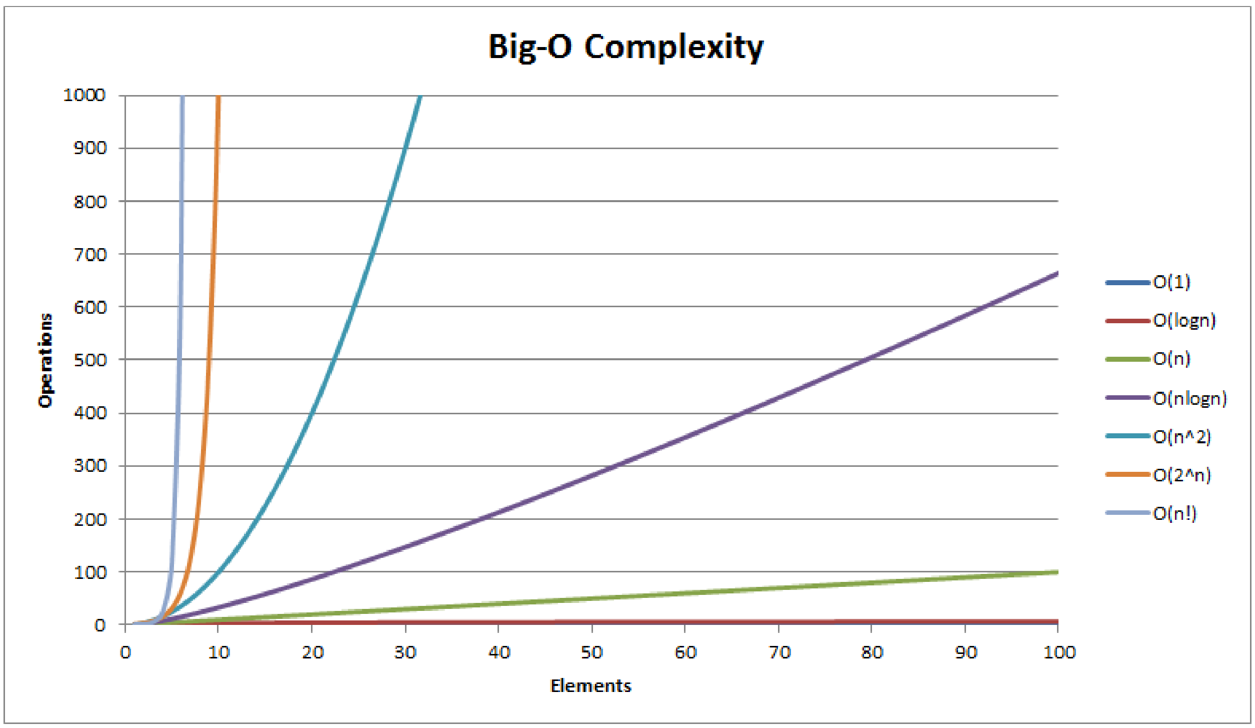

Photo Credit: https://medium.com/journey-of-one-thousand-apps/complexity-and-big-o-notation-in-swift-478a67ba20e7