**Computer Hardware and Performance**

1. All examples are in Ruby, but concept applies to all languages
2. A single memory location (bit) is equivalent to a single value on a Turing tape (a long tape that is basically binary that represents true or false, which is the minimum requirement to do any computation)
3. A Turing tape is like a cassette
4. RAM is 5-10 times faster than SSD, and thousands times faster than spinning hard disks
5. Computer memory is like a desk
   1. The table top is RAM, easy to access, but a gust of wind (power interruption) will blow it away
   2. The desk drawers are SSD/disk drive, slower to access but protected
6. Swapping is when your computer uses disk drive space to do RAM things, noticeably slows down comp (like a spinning wheel or application freeze)
7. What order does your solution operate at (Big-0 Notation, aka O(n))?
   1. NOTE - Y axis is time, X axis is number of times/tasks/data
   2. Order n or O(n)- linear relationship - double the items, double the time
      1. Examples:
         1. [1,2,3].include? 0
         2. [1,2,3,4,5,6].include? 0
   3. Order 1 or O(1) - the perfect state - no matter how many things there are, time is the same
      1. Examples:
         1. [1,2,3][0]
         2. {a:1, b:2, c:3}[:a]
   4. Order log n or O(log n) - early additions have high cost, and later additions have low cost
      1. Examples:
         1. Binary Search - like looking in phonebook. You slice the data at the mid point, check the data; if it’s before you eliminate the first half, if it’s after your eliminate the second half, then slice again (until you find it)
   5. Order n log n or O(n log n) - early additions have low costs, later additions have more cost
      1. Examples:
         1. Sorting - look at each item and arrange them compared to others, the memory locations are re-written as you go through the sort
   6. There are many more, google that shit
   7. Bad things to do:
      1. O(n) \* O(n) = O(n^2)
      2. Ex: Nesting a loop in a loop
8. Moore’s law - every 2 years, processing power will double for half the cost; **UNTIL RECENTLY** (mid 2000’s)
   1. More cores means do more things in parallel, opposed to doing things faster