Carter Watson
09/14/2023
CS5050 - Advanced Algorithms

Assignment 1

1. Fugaku
   instructions/second - 4.0 X 10^17
   Complexity - O(2^n)
   a. n = 100
      2^100 = 1.2676506 X 10^30 instructions
      (1.2676506 X 10^30) * (1 / (4.0 X 10^17)) = 3.1691265 X10^12 seconds
      (3.1691265 X10^12) / (3.154 X 10^9) = 1.004877473 X 10^3 centuries
   b. n = 1000
      2^1000 = 1.071509 X 10^301
      (1.071509 X 10^301) * (1 / (4.0 X 10^17)) = 2.678773 X 10^283 seconds
      (2.678773 X 10^283) / (3.154 X 10^9) = 8.49433346 X 10^273 centuries
2. Ordered small to large
   a. 2^500 smallest
   b. log(log n)^2
   c. log n
   d. log_4 n
   e. 2^(log n)
   f. (log n)^3
   g. n^(1/2)
   h. n log n
   i. n^2 (log n)^5
   j. n^3
   k. 2^n
   l. n!
3. f(n) = O(g(n))
   f(n) = omega(g(n))
   f(n) = theta(g(n))
   a. f(n) = theta(g(n)) - g(n) can grow faster or slower depending on constant
   b. f(n) = O(g(n)) - g(n) grows faster
   c. f(n) = omega(g(n)) - g(n) grows slower
   d. f(n) = O(g(n)) - g(n) grows faster
   e. f(n) = O(g(n)) - g(n) grows faster
   f. f(n) = theta(g(n)) - g(n) can grow faster or slower depending on constant