German University in Cairo
Department of Computer Science
Dr. Nourhan Ehab

CSEN 703 Analysis and Design of Algorithms, Winter Term 2022
Practice Assignment 2

Exercise 2-1 From CLRS (©MIT Press 2001)

Asymptotically rank the following functions:
$n, n^{1/2}, log(n), log(log(n)), log^2(n), (\frac{1}{3})^n, 4, (\frac{3}{2})^n, n!$

Exercise 2-2 From CLRS (©MIT Press 2001)

Explain why the statement: The running time of algorithm A is at least $O(n^2)$ is meaningless.

Exercise 2-3 From CLRS (©MIT Press 2001)

Prove that the running time of an algorithm is $\Theta(g(n))$ if and only if its worst-case running time is $O(g(n))$ and its best-case running time is $\Omega(g(n))$.

Exercise 2-4 From CLRS (©MIT Press 2001)

For every given $f(n)$ and $g(n)$ prove that $f(n) = \Theta(g(n))$

a) $g(n) = n^3$, $f(n) = 3n^3 + n^2 + n$

b) $g(n) = 2^n$, $f(n) = 2^{n+1}$

c) $g(n) = \ln(n)$, $f(n) = \log_{10}(n) + \log_{10}(\log_{10} n)$

Exercise 2-5

For every given $f(n)$ and $g(n)$ prove that $f(n) = o(g(n))$ or $f(n) = \omega(g(n))$

a) $f(n) = n^3$, $g(n) = n^2$

b) $f(n) = \log(n)$, $g(n) = \log^2(n)$

Exercise 2-6 From CLRS (©MIT Press 2001)

Let $f(n)$ and $g(n)$ be asymptotically non-negative functions. Using the basic definition of Θ-notation, prove that $max(f(n), g(n)) = \Theta(f(n) + g(n))$.

Exercise 2-7 From CLRS (©MIT Press 2001)

Show that for any real constants a and b, where b>0, $(n + a)^b = \Theta(n^b)$.

Exercise 2-8

Prove that, for $a, b \in \mathbb{R}$, $b > a \rightarrow a^n = o(b^n)$.