King Saud University
College of Computer and Information Sciences
Department of Computer Science

Design and Analysis of Algorithms (CSC311) – Spring 2017
Instructor: Prof. Mohamed Menai

Tutorial 2 (Asymptotic Notations)
Thu. Feb. 23rd, 2017

1. Use the definition of big-O to prove that $1^2 + 2^2 + \cdots + n^2$ is $O(n^3)$.
2. Use the definition of big-O to prove that $\dfrac{3n - 8 - 4n^3}{2n - 1}$ is $O(n^2)$.
3. Use the definition of big-O to prove that $1.2 + 2.3 + 3.4 + \cdots + (n-1).n$ is $O(n^3)$.
4. Show that $\sum_{j=1}^{n}(j^3 + j)$ is $O(n^4)$.
5. Show that $f(x) = (x+2)\log_2(x^2+1) + \log_2(x^3+1)$ is $O(x \log_2 x)$.
6. Prove that $\dfrac{x^3 + 7x^2 + 3}{2x + 1}$ is $\Theta(x^2)$.
7. Arrange the functions $n^{3/2}, \log(n^n), (n^{100})^n$ and $\log(n!)$ in a list so that each function is big-O of the next one.
8. Suppose you have two different algorithms for solving a problem. To solve a problem of size n, the first algorithm uses exactly $n\sqrt{n}$ operations and the second algorithm uses exactly $n^2\log(n)$ operations. As n grows, which algorithm uses fewer operations?
9. Find the best big-O notation to describe the time complexity of the following algorithms:
   (a) An algorithm that finds the average of n numbers.
   (b) A linear search of a list of size n (counting the number of comparisons).
   (c) A binary search of n elements.
   (d) An algorithm that prints all bit strings of length n.
   (e) An iterative algorithm to compute $n!$, (counting the number of multiplications).
10. Describe an algorithm that takes a list of n positive integers and finds the location of the last even integer in the list, and returns 0 if there are no even integers in the list. Give the best-case and worst-case running times of your algorithm in asymptotic notation.