CSCE 222-200 Discrete Structures for Computing

Fall 2024

Instructor: Dr. Jianer Chen   **Teaching Assistant:** Evan Kostov
Office: PETR 428   **Office:** EABC Cubicle 6
Phone: (979) 845-4259   **Phone:** (469) 996-5494
Email: chen@cse.tamu.edu   **Email:** evankostov@tamu.edu
Office Hours: T+R 2:00pm–3:30pm   **Office Hours:** MW 4:00pm-5:00pm

Assignment # 3
(Due October 10)

1. Give a big-O estimate for the number of number additions (i.e., the additions in the fourth statement $t = t + i + j$) in the following algorithm.

$t = 0;$
for $(i = 1; i \leq n; i++)$
    for $(j = 1; j \leq n; j++)$
        $t = t + i + j.$

2. Give a big-O estimate for the number of arithmetic operations (i.e., additions and multiplications) in the following algorithm. What is the value of t at the end of the algorithm?

$i = 1; \quad t = 0;$
while $(i \leq n)$
    $\{ t = t + i; \quad i = 2i \}$

3. How much time does an algorithm take to solve a problem of size n if this algorithm uses $2n^2 + 2^n$ operations, each requring 10^{-9} seconds, with each of the following values of n?

    a) 20   **b)** 50   **c)** 100   **d)** 200

4. Devise an algorithm that on an array $A[1..n]$ of n integers prints out (using a statement `print(i)`) all indices i such that $A[i] > A[1] + A[2] + \cdots + A[i-1]$. What is the time complexity of your algorithm in terms of big-O notation?

5. Devise an algorithm for finding the first and second largest elements in an array $A[1, , n]$ of n integers. What is the time complexity of your algorithm in terms of big-O notation?