COMP523 Tutorial 2

Coordinator: Aris Filos-Ratsikas Demonstrator: Michail Theofilatos

October 2, 2019

Problem 1

Solve the following recursive formulas.

A. $T(n) \leq T(n/2) + 4$

B. $T(n) \leq T(n/2) + 5n$

C. $T(n) \leq T(n/2) + 3n^2$

D. $T(n) \leq \frac{3}{2}T(n/2) + 1$

Sort the obtained formulas in terms of their asymptotic order.

Problem 2

Explain, step by step

A. how the MAJORITY algorithm will run on input 100011110.

B. how the INTEGERMULTIPLICATION algorithm will multiply the numbers 6 and 9.

Problem 3

Provide a (as-tight-as possible) bound for the asymptotic memory requirements of the following algorithms.

A. The MERGESORT algorithm.

B. The QUICKSORT algorithm where we take the element $A[n]$ as the pivot.

C. The MAJORITY algorithm.

D. The MAJORITYSORTED algorithm for a sorted array.

Problem 4

Prove that every binary tree of depth d has at most 2^d leaves.

Problem 5

Consider the search algorithm TernarySearch, which splits the array into three parts of size $n/3$ and then recursively calls itself on one of the three parts only, after determining which is the only part where the search element might lie, using a number of comparisons.

A. Write the pseudocode for the algorithm.

B. Provide a tight upper bound for the asymptotic running time of the algorithm, counting only the number of comparisons as operations.

C. In terms of the actual (not asymptotic) number of comparisons, which algorithm is faster? Explain your answer. You may ignore the comparisons needed to verify whether the size of the array is 1.

Problem 6

Consider the following problem: You are given an array $A[1, \ldots, n]$ consisting of n integer numbers and a target integer number x and you want to decide whether there exists a pair of integers i, j such that $1 \leq i, j \leq n$ and $A[i] + A[j] = x$. Design an algorithm that solves this problem in asymptotic running time $O(n \log n)$. It is sufficient to describe the algorithm (i.e., it is not necessary to write the pseudocode).