CS 477/677 Analysis of Algorithms

Spring 2024

Homework 2

Due date: February 13, 2024

1. (U & G-required) [30 points] Solve the following recurrences using the method of your choice.

a) $T(n) = 2T\left(\frac{n}{2}\right) + 5n^2$

b) $T(n) = 3T\left(\frac{n}{3}\right) + nlgn$

c) $T(n) = 7T\left(\frac{n}{3}\right) + n^3$

2. (U & G-required) [40 points] Solve the following recurrences using the method indicated:

a) [20 points] $T(n) = 2T\left(\frac{n}{4}\right) + n$ using the recursion tree method.

b) [20 points] Show by substitution that $T(n) = 2T\left(\frac{n}{4}\right) + 1$ is $O(n)$.

3. (U & G-required) [15 points] Consider the following recursive algorithm:

```
ALGORITHM Secret (A, lowIdx, highIdx, key)
// Input:
//   - An array A[1..n]  of integer numbers
//   - integers lowIdx, highIdx, key
// Variables:
//   - integers midIdx, retValue, temp1, temp2
if lowIdx == highIdx
     if A[lowIdx] == key
          return 1
     else
          return 0
else
     midIdx = ⌊(lowIdx+highIdx)/2⌋

     temp1 ← Secret (A, lowIdx, midIdx, key)
     temp2 ← Secret (A, midIdx+1, highIdx, key)
     return temp1 + temp2
```

a) [5 points] What does this algorithm return?

b) [10 points] Set up a recurrence relation for the algorithm's running time and solve it.

4. (U & G-required) [15 points]

Write pseudocode for a recursive algorithm that computes a^n using the recurrence formula $a^n = a * a^{n-1}$, where $a > 0$ and n is a positive integer.

5. (G-required) [20 points]

Set up and solve the recurrence relation for the running time of the algorithm from Problem 4.

Extra credit:

5. [20 points] Set up and solve a recurrence relation for the **number of additions** made by the following recursive algorithm for computing the sum of the first n even integers:

```
ALGORITHM SumEven(n)
    // Input: A nonnegative integer n
    // Output: The sum of the first even n integers
    if n = 1
        return 2
    else
        return SumEven(n-1) + n + n
```