CS385Q: Algorithms
Fall 2022
Homework2

Recurrence Relations

Due on: 10/28/2022 11:59 PM

1. (5*5 = 25 Points) Solve the following recurrence relations (using 5-step backward substitution method with a demonstration of all the steps):

a) $x(n) = x(n-1) + 5$ for $n > 1, \quad x(1) = 0$

b) $x(n) = 3x(n-1)$ for $n > 1, \quad x(1) = 4$

c) $x(n) = x(n-1) + n$ for $n > 0, \quad x(0) = 0$

d) $x(n) = x(n/2) + n$ for $n > 1, \quad x(1) = 1$ (solve for $n = 2^k$)

e) $x(n) = x(n/3) + 1$ for $n > 1, \quad x(1) = 1$ (solve for $n = 3^k$)

2. (10 Points) For each function below, compute the recurrence relation for its running time and then use the Master Theorem to find its complexity by specifying the different terms of the term explicitly:

```
int f(int arr[], int n) {
    if (n == 0) {
        return 0;
    }
    int sum = 0;
    for (int j = 0; j < n; ++j) {
        sum += arr[j];
    }
    return f(arr, n / 2) + sum + f(arr, n / 2);
}
```

```
void g(int n, int arrA[], int arrB[]) {
    if (n == 0) {
        return;
    }
    for (int i = 0; i < n; ++i) {
        for (int j = 0; j < n; ++j) {
            arrB[j] += arrA[i];
        }
    }
    g(n / 2, arrA, arrB);
}
```

3. (5*3 = 15 points) Use the Master Theorem to find the complexity of each of the following recurrence relations (show all the steps and the values of different terms to apply the theorem):

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

b) $T(n) = 4T\left(\frac{n}{2}\right) + n^2$

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