CS456 – Algorithm Design and Analysis - In-class discussion on 25th January 2024

Q1. Use recursion tree to solve each of the following recurrences.

a. $D(n) = 3D(n/3) + \sqrt{n}$

b. $E(n) = 3E(n/3) + n$

c. $F(n) = 3F(n/3) + n^2$

Q2. Use recursion tree to solve each of the following recurrences.

a. $M(n) = 2M(n/2) + O(n \log n)$

b. $N(n) = 2N(n/2) + O(n/\log n)$

c. $J(n) = J(n/2) + J(n/3) + J(n/6) + n$

Q3. Solve the following recurrence relation using Master's Theorem

a. $T(n) = 3T(n/2) + cn, \quad T(1) = 1$

b. $T(n) = 7T(n/2) + cn^2, \quad T(1) = 1$

c. $T(n) = 2T(n/4) + n^{0.51}$

d. $T(n) = 0.5T(n/2) + 1/n$ (master's theorem does work here. Why?)

Q4. Solve the following recurrences using the Recurrence Tree Method

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

2) $T(n) = 2T\left(\frac{n}{3}\right) + n$

3) $T(n) = 2T(n-1) + 1$

4) $T(n) = 3T\left(\frac{n}{4}\right) + n$

5) $T(n) = T\left(\frac{n}{4}\right) + T\left(\frac{3n}{4}\right) + n$

Figure 1

Q5. Find a recurrence relation for the program below and solve the recurrence relation to find the equivalent closed forms.

```
1.Algorithm printDStars(int n)
    if (n > 0)
        print("*d");
    printDStars(n-1)
    printDStars(n-1)
    EndAlgorithm
```

```
2. Algorithm printSStars(int n)
    if (n > 0)
        for (k ← 1; k < n; k=k*2)
            print("*s")
    printSStars(n-1)
    EndAlgorithm
```