Design and Analysis of Algorithms
Tutorial 2

童咏昕

北京航空航天大学 计算机学院

yxtong@buaa.edu.cn

Question 1

Give asymptotic upper bounds for T(n) by recursion tree approach. Make your bounds as tight as possible.

(a)

$$T(1) = 1$$
$$T(n) = T\left(\frac{n}{2}\right) + n \qquad if\ n > 1$$

(b)

$$T(1) = T(2) = 1$$
$$T(n) = T(n-2) + 1 \quad if\ n > 2$$

(c)

$$T(1) = 1$$
$$T(n) = T\left(\frac{n}{3}\right) + n \qquad if\ n > 1$$

Question 1

(d)

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

(e)

$$T(1) = 1$$
$$T(n) = 3T\left(\frac{n}{2}\right) + n^2 \qquad if\ n > 1$$

(f)

$$T(1) = 0, T(2) = 1$$
$$T(n) = T\left(\frac{n}{2}\right) + \log_2 n \qquad if\ n > 2$$

Question 2

Prove the following problems by induction.

(a) Given

$$T(1) = 1$$
$$T(n) = T\left(\frac{n}{2}\right) + n \qquad if\ n > 1$$

Prove $T(n) \leq c \cdot n$ for some c.

(b) Given

$$T(1) = T(2) = 1$$
$$T(n) = T(n-2) + 1 \qquad if\ n > 1$$

Prove $T(n) \leq c \cdot n$ for some c.

(c) Given

$$T(1) = 1$$
$$T(n) = T\left(\frac{n}{3}\right) + n \qquad if\ n > 1$$

Prove $T(n) \leq c \cdot n$ for some c.

Question 2

(d) Given

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

Prove $T(n) \le c_1 \cdot n^2 - c_2 \cdot n$ for some c_1 and c_2.

(e) Given

$$T(1) = 1$$
$$T(n) = 3T\left(\frac{n}{2}\right) + n^2 \qquad if\ n > 1$$

Prove $T(n) \le c_1 \cdot n^2$ for some c.

(f) Given

$$T(1) = 0, T(2) = 1$$
$$T(n) = T\left(\frac{n}{2}\right) + \log_2 n \qquad if\ n > 2$$

Prove $T(n) \le c \cdot \log^2 n$ for some c.

Question 3

Let A[1..n] be an array of positive integers.

Design a divide-and-conquer algorithm for computing the maximum value of $A[j] - A[i]$ with $j \geq i$.

Analyze your algorithm running time.