Technische Universität München
Fakultät für Informatik
Lehrstuhl für Effiziente Algorithmen
Prof. Dr. Harald Räcke
Chintan Shah

Wintersemester 2013
Assignment 2
28 October 2013

Efficient Algorithms and Datastructures I

Question 1 (10 Points)

1. Solve the following recurrence relation without using generating functions:

$$a_n = a_{n-1} + 2^{n-1} \text{ for } n \geq 1 \text{ with } a_0 = 2.$$

2. Give tight asymptotic upper and lower bounds for $T(n)$:

$$T(n) = T(n-1) + \log n.$$

Question 2 (5 Points)

Give tight asymptotic upper and lower bounds for $T(n)$:

$$T(n) = T\left(\frac{n}{2}\right) + T\left(\frac{n}{4}\right) + T\left(\frac{n}{8}\right) + n.$$

Question 3 (5 Points)

Given two $n \times n$ matrices A and B where n is a power of 2, we know how to find $C = A \cdot B$ by performing n^3 multiplications. Now let us consider the following approach. We partition A, B and C into equally sized block matrices as follows:

$$A = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} B = \begin{bmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{bmatrix} C = \begin{bmatrix} C_{11} & C_{12} \\ C_{21} & C_{22} \end{bmatrix}$$

Consider the following matrices:

$$\begin{aligned}
M_1 &= (A_{11} + A_{22})(B_{11} + B_{22}) \\
M_2 &= (A_{21} + A_{22})B_{11} \\
M_3 &= A_{11}(B_{12} - B_{22}) \\
M_4 &= A_{22}(B_{21} - B_{11}) \\
M_5 &= (A_{11} + A_{12})B_{22} \\
M_6 &= (A_{21} - A_{11})(B_{11} + B_{12}) \\
M_7 &= (A_{12} - A_{22})(B_{21} + B_{22})
\end{aligned}$$

Then,

$$\begin{aligned} C_{11} &= M_1 + M_4 - M_5 + M_7 \\ C_{12} &= M_3 + M_5 \\ C_{21} &= M_2 + M_4 \\ C_{22} &= M_1 - M_2 + M_3 + M_6 \end{aligned}$$

1. Convince yourself that the matrices C_{ij} evaluated as above are indeed correct. Don't write anything to prove this.
2. Design an efficient algorithm for multiplying two $n \times n$ matrices based on these facts. Analyze its running time.

Question 4 (10 Points)

Consider the following procedure:
RECURSIVE-SORT$(A, i, j)\{$
if $(A[i] > A[j])$ **then** swap $A[i] \leftrightarrow A[j]$
if $i + 1 \geq j$ **then return**
$k \leftarrow \lfloor (j - i + 1)/3 \rfloor$
RECURSIVE-SORT$(A, i, j - k)$
RECURSIVE-SORT$(A, i + k, j)$
RECURSIVE-SORT$(A, i, j - k)$
$\}$

1. Argue that *RECURSIVE-SORT*$(A, 1, n)$ correctly sorts a given array $A[1 \dots n]$.
2. Analyze the running time of *RECURSIVE-SORT* using a recurrence relation.