Universität Siegen
Lehrstuhl Theoretische Informatik
Markus Lohrey

Exercise 2

Task 1
Calculate $2063 \cdot 3201$ using the algorithm of Karatsuba. You do not have to use base 2.

Solution
For base b we have

$$rs = ACb^n + (A - B)(D - C)b^{\frac{n}{2}} + (BD + AC)b^{\frac{n}{2}} + BD.$$

$b = 10$, $n = 2^2 = 4$, $r = 2063 = A \mid B$, $A = 20$, $B = 63$, $s = 3201 = C \mid D$, $C = 32$, $D = 1$

$$\begin{aligned} rs &= (20 \cdot 32) \cdot 10^4 + (20 - 63)(1 - 32) \cdot 10^2 + (63 \cdot 1 + 20 \cdot 32) \cdot 10^2 + 63 \cdot 1 \\ &= 640 \cdot 10^4 + 1333 \cdot 10^2 + (63 + 640) \cdot 10^2 + 63 \\ &= 6603663 \end{aligned}$$

The second = makes use of the following 3 calculations:

(a) $20 \cdot 32$, $n = 2$, $A = 2$, $B = 0$, $C = 3$, $D = 2$

$$\begin{aligned} 20 \cdot 32 &= (2 \cdot 3) \cdot 10^2 + (2 - 0)(2 - 3) \cdot 10^1 + (0 \cdot 2 + 2 \cdot 3) \cdot 10^1 + 0 \cdot 2 \\ &= 600 - 20 + 60 = 640 \end{aligned}$$

(b) $(20 - 63)(1 - 32) = 43 \cdot 31$, $n = 2$, $A = 4$, $B = 3$, $C = 3$, $D = 1$

$$\begin{aligned} 43 \cdot 31 &= (4 \cdot 3) \cdot 10^2 + (4 - 3)(1 - 3) \cdot 10^1 + (3 \cdot 1 + 4 \cdot 3) \cdot 10^1 + 3 \cdot 1 \\ &= 1200 - 20 + 150 + 3 = 1333 \end{aligned}$$

(c) $63 \cdot 1$, $n = 2$, $A = 6$, $B = 3$, $C = 0$, $D = 1$

$$\begin{aligned} 63 \cdot 1 &= (6 \cdot 0) \cdot 10^2 + (6 - 3)(1 - 0) \cdot 10^1 + (3 \cdot 1 + 6 \cdot 0) \cdot 10^1 + 3 \cdot 1 \\ &= 30 + 30 + 3 = 63 \end{aligned}$$

Task 2
Use the algorithm of Strassen to calculate the following matrix product:

$$\begin{pmatrix} 3 & -2 \\ 1 & 0 \end{pmatrix} \cdot \begin{pmatrix} 1 & 2 \\ -1 & 1 \end{pmatrix}$$

Solution

- $M_1 = (A_{12} - A_{22})(B_{21} + B_{22}) = (-2 - 0)(-1 + 1) = 0$
- $M_2 = (A_{11} + A_{22})(B_{11} + B_{22}) = (3 + 0)(1 + 1) = 6$
- $M_3 = (A_{11} - A_{21})(B_{11} + B_{12}) = (3 - 1)(1 + 2) = 6$
- $M_4 = (A_{11} + A_{12})B_{22} = (3 + (-2)) \cdot 1 = 1$
- $M_5 = A_{11}(B_{12} - B_{22}) = 3(2 - 1) = 3$
- $M_6 = A_{22}(B_{21} - B_{11}) = 0(-1 - 1) = 0$
- $M_7 = (A_{21} + A_{22})B_{11} = (1 + 0) \cdot 1 = 1$
- $C_{11} = M_1 + M_2 - M_4 + M_6 = 0 + 6 - 1 + 0 = 5$
- $C_{12} = M_4 + M_5 = 1 + 3 = 4$
- $C_{21} = M_6 + M_7 = 0 + 1 = 1$
- $C_{22} = M_2 - M_3 + M_5 - M_7 = 6 - 6 + 3 - 1 = 2$

Hence we have

$$\begin{pmatrix} 3 & -2 \\ 1 & 0 \end{pmatrix} \cdot \begin{pmatrix} 1 & 2 \\ -1 & 1 \end{pmatrix} = \begin{pmatrix} C_{11} & C_{12} \\ C_{21} & C_{22} \end{pmatrix} = \begin{pmatrix} 5 & 4 \\ 1 & 2 \end{pmatrix}$$

Task 3

Professor Caesar wishes to develop a matrix-multiplication algorithm that is asymptotically faster than Strassen's algorithm. His algorithm will use the divide-and-conquer method, dividing each matrix into pieces of size $n/4 \times n/4$, and the divide and combine steps together will take $\Theta(n^2)$ time. He needs to determine how many subproblems his algorithm has to create in order to beat Strassen's algorithm. If his algorithm creates a subproblems then the recurrence for the running time $T(n)$ becomes $aT(n/4) + \Theta(n^2)$. What is the largest integer value for a for which Professor Caesar's algorithm would be asymptotically faster than Strassen's algorithm?

Solution

Strassen's algorithm has a running time of $\Theta(n^{\frac{\log(7)}{\log(2)}}) = \Theta(n^{2.807\cdots})$. Using the Master Theorem for Caesar's algorithm, we have $b^c = 4^2 = 16$. Depending on a, it has the following running time:

- $\Theta(n^2)$ if $a < 16$ (which is better than Strassen's algorithm),
- $\Theta(n^2 \log n)$ if $a = 16$ (which is also better than Strassen's algorithm)

- $\Theta\left(n^{\frac{\log(a)}{\log(4)}}\right)$ if $a > 16$. In this case, we want that $\frac{\log(a)}{\log(4)} < \frac{\log(7)}{\log(2)}$.
  Therefore, $\frac{\log(a)}{\log(7)} < \frac{\log(4)}{\log(2)} = 2$ and thus $a < 49$ $(17 \leq a \leq 48)$.

Hence, the largest integer for a is 48.

Task 4
Show that a binary tree with N leaves has at least height $\log_2(N)$.

Solution
The set of binary trees $\mathcal{T}$ is defined as follows: $() \in T$ is a binary tree, and if $t_1, t_2 \in T$, then $(t_1, t_2) \in \mathcal{T}$. The height $h\colon \mathcal{T} \to \mathbb{N}$ is defined as: $h() = 1$, $h(t_1, t_2) = 1 + \max\{h(t_1), h(t_2)\}$. The number of leaves $\ell\colon \mathcal{T} \to \mathbb{N}$ is defined as $\ell() = 1$ and $\ell(t_1, t_2) = \ell(t_1) + \ell(t_2)$. We now want to show that for each $t \in \mathcal{T}$ it holds that $h(t) \geq \log_2(\ell(t))$.

- For $t = ()$ we have $h(t) = 1 \geq 0 = \log_2(1) = \log_2(\ell(t))$.
- For $t = (t_1, t_2)$ we have

$$\begin{aligned}
h(t) &= 1 + \max\{h(t_1), h(t_2)\} \\
&\geq 1 + \max\{\log_2(\ell(t_1)), \log_2(\ell(t_2))\} \\
&= 1 + \log_2(\max\{\ell(t_1), \ell(t_2)\}) \\
&= \log_2(2 \max\{\ell(t_1), \ell(t_2)\}) \\
&\geq \log_2(\ell(t_1) + \ell(t_2)) \\
&= \log_2(\ell(t))
\end{aligned}$$

We made use of the following statements: For all $x, y \geq 0$:

- $\max\{\log(x), \log(y)\} = \log(\max\{x, y\})$. This is true, because log is monotone.
- $2\max\{x, y\} \geq x + y$. If $x \leq y$, then $x + y \leq y + y = 2y = 2\max\{x, y\}$. The case $y < x$ is similar.

Task 5
Sort the array $[2, 8, 13, 5, 7, 16, 3, 12]$ using Quicksort.

Solution
quicksort$(1, 8)$: $p = 4$ (index of median of $A[\ell]$, $A[(\ell + r) \text{ div } 2]$, $A[r]$), partition$(1, 8, 4)$: swap$(4, 8)$ (pivot), swap$(1, 1)$, swap$(2, 7)$, swap$(3, 8)$ (pivot), $[2, 3, 5, 12, 7, 16, 8, 13]$, $m = 3$

- quicksort$(1, 2)$: $p = 1$, swap$(1, 2)$, swap$(1, 2)$, $[1, 2]$, $m = 1$
  - quicksort$(1, 0)$
  - quicksort$(2, 2)$

- quicksort$(4, 8)$: $p = 8$, partition$(4, 8, 8)$: swap$(8, 8)$ (pivot), swap$(4, 4)$, swap$(5, 5)$, swap$(6, 7)$, swap$(7, 8)$ (pivot), $[2, 3, 5, 12, 7, 8, 13, 16]$, $m = 7$
  - quicksort$(4, 6)$: $p = 6$, partition$(4, 6, 6)$: swap$(6, 6)$ (pivot), swap$(4, 5)$, swap$(5, 6)$, $[2, 3, 5, 7, 8, 12, 13, 16]$, $m = 5$
    - quicksort$(4, 4)$
    - quicksort$(6, 6)$
  - quicksort$(8, 8)$