Universität Siegen
Lehrstuhl Theoretische Informatik
Markus Lohrey

Exercise 2

Task 1
Sort the array $[4, 19, 8, 1, 18, 7, 11, 13]$ using Mergesort.

Solution
mergesort$(1, 8)$, $m = 4$

- mergesort$(1, 4)$, $m = 2$
  - mergesort$(1, 2)$, $m = 1$, merge$(1, 1, 2)$, $[4, 19]$
  - mergesort$(3, 4)$, $m = 3$, merge$(3, 3, 4)$, $[1, 8]$
  - merge$(1, 2, 4)$, $[1, 4, 8, 19]$
- mergesort$(5, 8)$, $m = 6$
  - mergesort$(5, 6)$, $m = 5$, merge$(5, 5, 6)$, $[7, 18]$
  - mergesort$(7, 8)$, $m = 7$, merge$(7, 7, 8)$, $[11, 13]$
  - merge$(5, 6, 8)$, $[7, 11, 13, 18]$

merge$(1, 4, 8)$, $[1, 4, 7, 8, 11, 13, 18, 19]$
Using the array-box representation (as dicussed in the exercise session) is also sufficient.

Task 2
Calculate $1295 \cdot 4077$ using the algorithm of Karatsuba. You do not have to use base 2.

Solution
For base b we have

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

$b = 10$, $n = 2^2 = 4$, $r = 1295 = A \mid B$, $A = 12$, $B = 95$, $s = 4077 = C \mid D$, $C = 40$, $D = 77$

$$\begin{aligned} rs &= (12 \cdot 40) \cdot 10^4 + (12 - 95)(77 - 40) \cdot 10^2 + (95 \cdot 77 + 12 \cdot 40) \cdot 10^2 + 95 \cdot 77 \\ &= 480 \cdot 10^4 - 3071 \cdot 10^2 + (7315 + 480) \cdot 10^2 + 7315 \\ &= 5279715 \end{aligned}$$

The second $=$ makes use of the following 3 calculations:

(a) $12 \cdot 40,\ n = 2,\ A = 1,\ B = 2,\ C = 4,\ D = 0$

$$\begin{aligned} 12 \cdot 40 &= (1 \cdot 4) \cdot 10^2 + (1 - 2)(0 - 4) \cdot 10^1 + (2 \cdot 0 + 1 \cdot 4) \cdot 10^1 + 2 \cdot 0 \\ &= 400 + 40 + 40 = 480 \end{aligned}$$

(b) $(12 - 95)(77 - 40) = -83 \cdot 37,\ n = 2,\ A = 8,\ B = 3,\ C = 3,\ D = 7$

$$\begin{aligned} 83 \cdot 37 &= (8 \cdot 3) \cdot 10^2 + (8 - 3)(7 - 3) \cdot 10^1 + (3 \cdot 7 + 8 \cdot 3) \cdot 10^1 + 3 \cdot 7 \\ &= 2400 + 200 + 450 + 21 = 3071 \end{aligned}$$

(c) $95 \cdot 77,\ n = 2,\ A = 9,\ B = 5,\ C = 7,\ D = 7$

$$\begin{aligned} 95 \cdot 77 &= (9 \cdot 7) \cdot 10^2 + (9 - 5)(7 - 7) \cdot 10^1 + (5 \cdot 7 + 9 \cdot 7) \cdot 10^1 + 5 \cdot 7 \\ &= 6300 + 980 + 35 = 7315 \end{aligned}$$

Task 3
Use the algorithm of Strassen to calculate the following matrix product:

$$\begin{pmatrix} 1 & -4 \\ 3 & 2 \end{pmatrix} \cdot \begin{pmatrix} 0 & 8 \\ -5 & 1 \end{pmatrix}$$

Solution

- $M_1 = (A_{12} - A_{22})(B_{21} + B_{22}) = (-4 - 2)(-5 + 1) = 24$
- $M_2 = (A_{11} + A_{22})(B_{11} + B_{22}) = (1 + 2)(0 + 1) = 3$
- $M_3 = (A_{11} - A_{21})(B_{11} + B_{12}) = (1 - 3)(0 + 8) = -16$
- $M_4 = (A_{11} + A_{12})B_{22} = (1 + (-4)) \cdot 1 = -3$
- $M_5 = A_{11}(B_{12} - B_{22}) = 1(8 - 1) = 7$
- $M_6 = A_{22}(B_{21} - B_{11}) = 2(-5 - 0) = -10$
- $M_7 = (A_{21} + A_{22})B_{11} = (3 + 2) \cdot 0 = 0$
- $C_{11} = M_1 + M_2 - M_4 + M_6 = 24 + 3 - (-3) + (-10) = 20$
- $C_{12} = M_4 + M_5 = -3 + 7 = 4$
- $C_{21} = M_6 + M_7 = -10 + 0 = -10$
- $C_{22} = M_2 - M_3 + M_5 - M_7 = 3 - (-16) + 7 - 0 = 26$

Hence we have
$$\begin{pmatrix} 1 & -4 \\ 3 & 2 \end{pmatrix} \cdot \begin{pmatrix} 0 & 8 \\ -5 & 1 \end{pmatrix} = \begin{pmatrix} C_{11} & C_{12} \\ C_{21} & C_{22} \end{pmatrix} = \begin{pmatrix} 20 & 4 \\ -10 & 26 \end{pmatrix}$$

Task 4

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 $n/8 \times n/8$, 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 $aT(n/8) + \Theta(n^2)$. 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 $b^c = 8^2 = 2^6 = 64$. Depending on a, it has the following running time:

- $\Theta(n^2)$ if $a < 64$ (which is better than Strassen's algorithm),
- $\Theta(n^2 \log n)$ if $a = 64$ (which is also better than Strassen's algorithm)
- $\Theta\left(n^{\frac{\log(a)}{\log(8)}}\right)$ if $a > 64$. In this case, we want that $\frac{\log(a)}{\log(8)} < \frac{\log(7)}{\log(2)}$.
  Therefore, $\frac{\log(a)}{\log(7)} < \frac{\log(8)}{\log(2)} = 3$ and thus $a < 7^3 = 343$ $(65 \leq a \leq 342)$.

Hence, the largest integer for a is 342.

Task 5

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.