CONCORDIA UNIVERSITY

DEPARTMENT OF COMPUTER SCIENCE AND SOFTWARE ENGINEERING

COMP 6651: Algorithm Design Techniques

Practice Exercises with their Solutions

B. JAUMARD

Fall 2019

Acknowledgments

I would like to thank the graders of the algorithm design course I have taught over the years under different names (Discrete Mathematics or Algorithm Design) and who have helped a lot in the writing and the checking of several solutions of the exercises of this exercise compendium. In particular, I would like to thank Christophe Meyer, Anh Hai Hoang, Minh Bui, Mahdi Negahishirazi and Kia Babashahi Ashtiani for their help in proofreading the solutions and in tracking the typos.

Contents

Chapter 1

Asymptotic Notations

Exercise 1.1

Compare the following pairs of functions in terms of order magnitude. In each case, say whether $f(n) = O(g(n))$, $f(n) = \Omega(g(n))$, and/or $f(n) = \Theta(g(n))$. Justify your answers.

	$f(n)$	$g(n)$
a.	n	$\log n$
b.	$\log n$	$\log(4n^2 + 2)$
c.	$100n + \log n$	$n + (\log n)^2$
d.	$\log n$	$\log(n^2)$
e.	$\dfrac{n^2}{\log n}$	$n(\log n)^2$
f.	$(\log n)^{\log n}$	$\dfrac{n}{\log n}$

Comment on the proposed solutions and proofs. My objective when writing the proofs was to illustrate different types of proofs, so that you can use the type of proofs you are the most familiar with. However, all questions could have been answered using the limit properties that we saw during Lecture 1. For some questions, I have proposed two different proofs.

a. $\quad f(n) = n \quad$ and $\quad g(n) = \log n$

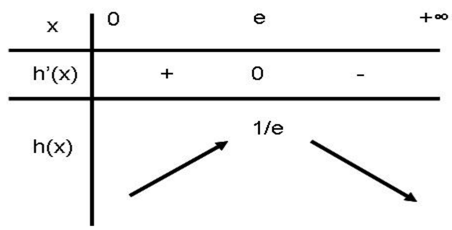
Answer: $\quad f(n) \neq O(g(n)); f(n) = \Omega(g(n)); f(n) \neq \Theta(g(n)).$

Solution 1

- $f(n) \neq O(g(n))$
  $\lim_{n \to \infty} \dfrac{f(n)}{g(n)} = \lim_{n \to \infty} \dfrac{n}{\log n} = +\infty.$
  Therefore $\exists N \; \forall C \; n \geq N \Rightarrow f(n) = n > Cg(n) = C \log n.$
- $f(n) = \Omega(g(n))$
  We first study the behavior of function f (i.e., monotonicity) using the sign of its derivative.

We will see that it will allow us to prove that $h(x) = \dfrac{\log x}{x} \leq 1$ for $x \geq 1$
Remember that $(\ln x)' = \dfrac{1}{x}$ and that $(\log x)' = \dfrac{1}{(\ln 10)x}$
The first derivative of $h(x)$ is written as follows:

$$h'(x) = \frac{(\log x)'}{x} - \frac{\log x}{x^2} = \frac{1 - \ln x}{x^2 \ \ln 10}.$$



Since $h(e) = 1/e < 1$, and as $h(x)$ decreases starting from $x = e$, then $h(x) \leq 1$ for $x \geq 1$.
We start use the mathematical definition of the Ω notation.
Choosing $C = 1$ and $N = 1$, we have: $\exists C, \exists N, \, n \geq N \Rightarrow f(n) = n \geq Cg(n) = C \log n$

- $f(n) \neq \Theta(g(n))$ since $f(n) \neq O(g(n))$

Solution 2

$$\lim_{n\to\infty} \frac{f(n)}{g(n)} = \lim_{n\to\infty} \frac{n}{\log n} = +\infty.$$

According to the results of Slide # 16, $f(n) = \Omega(g(n))$, but $f(n) \neq \Theta(g(n))$.

b. $f(n) = \log n$ and $g(n) = \log(4n^2 + 2)$

Answer : $f(n) = O(g(n)); f(n) = \Omega(g(n)); f(n) = \Theta(g(n))$

Solution 1

- $f(n) = O(g(n))$
  for $n \geq 0, \quad n \leq 4n^2 + 2$
  since log is an increasing function, then $\log n \leq \log(4n^2 + 2)$ for $n \geq 1$
  The solution is found for $C = N = 1$.

- $f(n) = \Omega(g(n))$
  For $n \geq 5$, it can be easily proved that $n^3 \geq 4n^2 + 2$. Since log is an increasing function, then $\log n^3 = 3 \log n \geq \log(4n^2 + 2)$ for $n \geq 5$. Taking $C = 1/3$ and $N = 5$, we get the sought solution.

  Hint (how to get the idea of the proof): In order to prove that $f(n) = \Omega(g(n))$, first go back to the definition of Ω. It tells you that you need to find two constant values (there may be several possible ones, but for the purpose of what you want to prove, finding one value that works, it is enough), C and N such that

  $$\forall n \geq N, \qquad f(n) = \log n \geq Cg(n) = C \log(4n^2 + 2).$$

  Then the first question is how to get C such that $\log n \geq C \log(4n^2 + 2)$.
  Remember that $\log a^b = b \log a$.
  If you want $\log n \geq C \log(4n^2 + 2)$, you need to increase (with the help of C) the degree of the polynomial inside the log function in the left hand side of the inequality. This is the case if you select $C = 1/3$ (of course any value larger than $1/3$ works as well).

- $f(n) = \Theta(g(n))$, since $f(n) = O(g(n))$ and $f(n) = \Omega(g(n))$

Solution 2

Let us compute:

$$\lim_{n \to +\infty} \frac{f(n)}{g(n)} = \lim_{n \to +\infty} \frac{\log n}{\log(4n^2 + 2)} = \lim_{n \to +\infty} \frac{\log n}{\log(4n^2)} = \lim_{n \to +\infty} \frac{\log n}{2 \log(2n)} = \text{ Constant value.}$$

Using the results we saw during Lecture 1 on the limits, we conclude that $f(n) = \Theta(g(n))$, and therefore that $f(n) = O(g(n))$ and $f(n) = \Omega(g(n))$.

c. $\quad f(n) = 100n + \log n \quad$ and $\quad g(n) = n + (\log n)^2$

Answer: $\quad f(n) = O(g(n)) = \Omega(g(n)) = \Theta(g(n))$.

- $f(n) = O(g(n))$
  $\forall n \geq 10 \quad \log n \leq (\log(n))^2$
  thus $\forall n \geq 10 \quad 100n + \log n \leq 100n + (\log(n))^2 \leq 100(n + (\log(n))^2)$
  which could be written $\forall n \geq 10 \quad f(n) = 100n + \log n \leq 100(n + (\log(n))^2) = 100g(n)$
  result found by taking $N = 10$ and $C = 100$.

- $f(n) = \Omega(g(n))$ (1st method)
  Theorem $\forall c > 0 \quad \forall a > 1 \quad \forall f(n)$ increasing $\quad (f(n))^c = O(a^{f(n)})$
  Taking $a = 10$ and $c = 2$; $f(n) = \log n \quad (\log n)^2 = O(10^{\log n}) = O(n)$

From the definition of $O(.)$, we deduce :

$$\exists N, \exists C \text{ such that } (\log n)^2 \leq Cn \quad \forall n \geq N.$$

It follows that:

$$\exists N, \forall n \geq N, \exists C \text{ such that } (\log n)^2 + n \leq (C+1)n.$$

Consequently,

$$\exists N', \exists C, \forall n \geq N' \quad (\log n)^2 + n \leq \frac{(C+1)}{100}(100n + \log n),$$

with $N' = \max\{1, N\}$
Taking $C' = \frac{100}{C+1}$, we get :
$\exists N'$, $\exists C$, such that $\forall n \geq N' \Rightarrow 100n + \log n \geq C'(n + (\log n)^2)$

- $f(n) = \Omega(g(n))$ (2nd method)
  Using what we have already proved in question a), we get

  $$\forall n \geq 1 \quad \sqrt{n} \geq \log(\sqrt{n}) = 1/2 \times \log n$$

  as well as

  $$\forall n \geq 1 \quad 2\sqrt{n} \geq \log n$$

  Elevating the two members to their square, (both members of the inequality are positive), we get

  $$\forall n \geq 1 \quad 4n \geq (\log n)^2$$

  On the other hand: $100n + \log n \geq 100n$, $\forall n \geq 1$.
  Thus $\forall n \geq 1$, $\quad 100n + \log n \geq n + 4n \geq n + (\log n)^2$. Taking $C = 1$, $N = 1$, we get the sought solution.

- $f(n) = \Theta(g(n))$, since $f(n) = O(g(n))$ and $f(n) = \Omega(g(n))$

d. $\quad f(n) = \log n \quad$ and $\quad g(n) = \log(n^2)$

Answer: $\quad f(n) = O(g(n)); f(n) = \Omega(g(n)); f(n) = \Theta(g(n)).$

- $f(n) = O(g(n)) \quad N = 1, C = 2$
- $f(n) = \Omega(g(n)) \quad N = 1, C = 0.5$

e. $\quad f(n) = \dfrac{n^2}{\log n} \quad$ and $\quad g(n) = n(\log n)^2$

Answer: $\quad f(n) \neq O(g(n)); f(n) = \Omega(g(n)); f(n) \neq \Theta(g(n)).$

- $f(n) \neq O(g(n))$

$$\lim_{n\to\infty} \frac{f(n)}{g(n)} = \lim_{n\to\infty} \frac{n}{(\log n)^3} = +\infty.$$

- $f(n) = \Omega(g(n))$
  Let $c = 3$ and $a = 10$, $f(n) = \log n$ in the previously defined Theorem: $(\log n)^3 = O(n)$
  We deduce that: $\exists C$, $\exists N$, $\forall n \geq N$, $\quad (\log n)^3 \leq Cn$
  It then follows: $\exists C$, $\exists N' = N$ such that $\forall n \geq N' \quad n(\log n)^2 = n\frac{\log n^3}{\log n} \leq C\frac{n^2}{\log n}$

- $f(n) \neq \Theta(g(n))$ since $f(n) \neq O(g(n))$

f. $\quad f(n) = (\log n)^{\log n} \quad$ and $\quad g(n) = \dfrac{n}{\log n}$

Answer: $\quad f(n) \neq O(g(n)); f(n) = \Omega(g(n)); f(n) \neq \Theta(g(n))$.

- $f(n) \neq O(g(n))$
  Let $n \geq 10^{10} \iff \log n \geq 10$, we deduce

$$\frac{f(n)}{g(n)} \geq \frac{10^{\log n}}{\dfrac{n}{\log n}} = \log n$$

  It then follows, $\lim_{n->\infty} \frac{f(n)}{g(n)} = +\infty$

- $f(n) = \Omega(g(n)) \quad C = 1,\ N = 10^{10}$

- $f(n) \neq \Theta(g(n))$ since $f(n) \neq O(g(n))$

Exercise 1.2

Give asymptotic upper bounds for $T(n)$ under the following recurrences. Make your bounds as tight as possible. You need to justify your answer.

1. $T(1) = 1; T(n) = 4T(n/4) + 100n$ *for* $n > 1$.
2. $T(1) = 1; T(n) = T(n-1) + \log n$ *for* $n > 1$.

Solution

1. $O(n \log n)$
2. $O(n \log n)$
   Observe that:

$$\begin{array}{lll} T(n) & = T(n-1) + \log n & \\ T(n-1) & = T(n-2) + \log(n-1) & \leq T(n-2) + \log n \\ T(n-2) & = T(n-3) + \log(n-2) & \leq T(n-3) + \log n \\ \ldots & & \\ T(2) & = T(1) + \log 2 & \leq 1 + \log n \end{array}$$

Therefore

$$T(n) \leq 1 + n \log n \qquad \rightsquigarrow \qquad T(n) = O(n \log n)$$

Chapter 2

Recurrence Relations

Exercise 2.1

The ancient Greeks were very interested in sequences resulting from geometric shapes such as the following triangular numbers:

(i) For $n \geq 1$, the nth triangular term t_n is defined by: $t_n = 1 + 2 + \cdots + n = n(n+1)/2$. Find and solve a recurrence relation for $s_n, n \geq 1$, where $s_n = t_1 + t_2 + \cdots + t_n$, the sum of the first n triangular numbers.

In an organic laboratory, Kelsey synthesizes a crystalline structure that is made up of 10,000,000 triangular layers of atoms. The first layer of the structure has one atom, the second layer has three atoms, the third one has 6 atoms and, in general, the nth layer has $1 + 2 + \ldots n = t_n$ atoms.

(ii) How many atoms are there in these crystalline structures of Kelsey?

(iii) How many atoms are packed between the 10,000th and 100,000th layer?

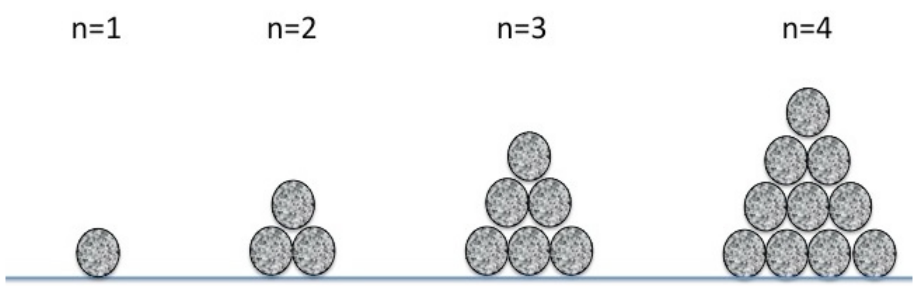


Figure 2.1: Geometric Shapes

Solution

(i) $s_n = s_{n-1} + t_n = s_{n-1} + \dfrac{n(n+1)}{2} \quad \rightsquigarrow \quad s_n = s_{n-1} + \dfrac{n(n+1)}{2}.$

General solution of the homogeneous recurrence relation: $\alpha - 1 = 0 \Rightarrow \alpha = 1$.

$$s_n^{(h)} = C_1(1)^n = C_1.$$

Particular solution of the non homogeneous recurrence relation: $p(n) = An^3 + Bn^2 + Cn + D$ as 1 is a root of multiplicity 1 of the characteristic equation.

From the definition of s_n, we know that: $s_n - s_{n-1} = t_n = \dfrac{n(n+1)}{2}$, therefore the same relation is valid for the particular solution, i.e., $p(n) - p(n-1) = \dfrac{n(n+1)}{2}$.
It follows that:

$$p(n) - p(n-1) = An^3 + Bn^2 + Cn + D - A(n-1)^3 - B(n-1)^2 - C(n-1) - D = \frac{n(n+1)}{2}.$$

After some algebraic manipulations, it leads to:

$$(B + 3A - B)n^2 + (C - 3A + 2B - C)n + D + A - B + C = \frac{1}{2}n^2 + \frac{1}{2}n.$$

Consequently,

$$3A = \frac{1}{2} \quad ; \quad -3A + 2B = \frac{1}{2} \quad ; \quad D + A - B + C = 0.$$

It follows that:

$$s_n^{(p)} = \frac{n^3}{6} + \frac{n^2}{2} - \frac{n}{3}.$$

General solution of the non homogeneous recurrence relation: $s_n = s_n^{(h)} + s_n^{(p)} = C_1 + \dfrac{n^3}{6} + \dfrac{n^2}{2} + \dfrac{n}{3}$.
C_1 can be identified using that $s_1 = 1$. It follows that $C_1 = 0$ then:

$$s_n = \frac{n^3}{6} + \frac{n^2}{2} + \frac{n}{3}.$$

(ii) $n = 10^7 \Rightarrow s_n = \dfrac{10^{21}}{6} + \dfrac{10^{14}}{2} + \dfrac{10^3}{3} \approx \dfrac{10^{21}}{6} \approx 1.67 \times 10^{20}$

(iii) s_n: the number of atoms from layer 1 to layer n, t_n: the number of atoms at layer n

$$s_{]10^4,10^5[} = s_{10^5} - t_{10^5} - s_{10^4} = \frac{10^{15} - 10^{12}}{6} \approx 1.665 \times 10^{14}$$

Exercise 2.2

Solve the following recurrence relations:

(i) $3a_{n+1} - 4a_n = 0, \quad n \geq 0, a_1 = 5$

(ii) $a_n = 5a_{n-1} + 6a_{n-2}, \quad n \geq 2, a_0 = 1, a_1 = 3$

(iii) $a_{n+2} + a_n = 0, \quad n \geq 0, a_1 = 3$

(iv) $a_n = 3a_{n/2} + n, \quad n = 2^k \geq 0, a_1 = 0$

(v) $a_n = a_{n-1} + n(n-1), \quad n \geq 1, a_0 = 2$

Solution

(i) $a_{n+1} = \frac{4}{3}a_n \rightsquigarrow$ root of the characteristic equation is equal to $r = \frac{4}{3} \quad \rightsquigarrow \quad a_n = C(\frac{4}{3})^n$
$a_1 = 5 = C\frac{4}{3} \quad \rightsquigarrow \quad C = \frac{15}{4} \quad \rightsquigarrow \quad a_n = \left(\frac{4}{3}\right)^n \times \frac{15}{4}$

(ii) $a_n = 5a_{n-1} - 6a_{n-2} = 0$
$\alpha^2 - 5\alpha - 6 = 0 \Rightarrow \alpha_1 = -1, \alpha_2 = 6$
$a_n = C_1(-1)^n + C_2 6^n$
$a_0 = C_1 + C_2 = 1, a_1 = -C_1 + 6C_2 = 3 \Rightarrow C_1 = 3/7; C_2 = 4/7$ $a_n = \frac{3}{7}(-1)^n + \frac{4}{7}6^n$

(iii) $a_{n+2} + a_n = 0$
$\alpha^2 + 1 = (\alpha - i)(\alpha + i) = 0 \Rightarrow \alpha_1 = i = \cos(\frac{\pi}{2}) + i\ \sin(\frac{\pi}{2}); \alpha_2 = -i = \cos(\frac{\pi}{2}) - i\ \sin(\frac{\pi}{2});$
$a_n = C_1\left(\cos(\frac{\pi}{2}) + i\ \sin(\frac{\pi}{2})\right)^n + C_2\left(\cos(\frac{\pi}{2}) - i\ \sin(\frac{\pi}{2})\right)^n$
$a_0 = C_1 + C_2 = 0; a_1 = i(C_1 - C_2) = 3 \Rightarrow C_1 = \frac{3}{2i}; C_2 = -\frac{3}{2i}$
$a_n = \frac{3}{2i}\left(\cos(\frac{\pi}{2}) + i\ \sin(\frac{\pi}{2})\right)^n - \frac{3}{2i}\left(\cos(\frac{\pi}{2}) - i\ \sin(\frac{\pi}{2})\right)^n$

Remember De Moivre's formula: $(\cos\theta + i\sin\theta)^n = \cos n\theta + i\sin n\theta$.

It follows that:

$$a_n = 3\sin\frac{n\pi}{2}.$$

(iv) $a_n = 3a_{n/2} + n$

- Change of variables to linearize the recurrence relation:
  $n = 2^k, d_k = a_{2^k} \Rightarrow d_k = 3d_{k-1} + 2^k$

- Solution of the homogeneous recurrence relation:
  $d_k^{(h)} = C_1 3^k$
- Particular solution of the non homogeneous relation:
  $d_k^{(p)} = C_2 a^k, d_k^{(p)} = 3d_{k-1}^{(p)} + 2^k$ take $a = 2 \Rightarrow C_2 = -2; d_k^{(p)} = -2^{k+1}$
- General solution of the non homogeneous recurrence relation:
  $d_k = d_k^{(h)} + d_k^{(p)} = C_1 3^k - 2^{k+1}; d_0 = 0 \Rightarrow C_1 = 2; d_k = 2 \times 3^k - 2^{k+1}$
  $a_n = 2 \times 3^{\log_2 n} - 2^{(\log_2 n + 1)} = 2 \times n^{\log_2 3} - 2 \times n$

(v) $a_n = a_{n-1} + n(n-1)$
Solution of the homogeneous recurrence relation: $a_n^{(h)} = a_{n-1}^{(h)}$, characteristic equation $r - 1 = 0; r = 1; \Rightarrow a_n^{(h)} = C_1$
Particular solution of the non homogeneous relation: The degree of the particular solution is 3: $a_n^{(p)} = An^3 + Bn^2 + Cn + D$, put back to the recurrence equation:

$$An^3 + Bn^2 + Cn + D = A(n-1)^3 + B(n-1)^2 + C(n-1) + D + n^2 - n$$

$$\Rightarrow (-3A+1)N^2 + (3A-2B)n + (-A+B-C) \equiv 0$$

$$\Rightarrow A = 1/3; B = 0; C = -1/3$$

General solution of the non homogeneous recurrence relation:
$a_n = a_n^{(p)} + a_n^{(h)} = \frac{1}{3}n^3 - \frac{1}{3}n + C_1; \quad a_0 = 2 \quad \rightsquigarrow \quad C_1 = 2.$

$$a_n = \frac{1}{3}n^3 - \frac{1}{3}n + 2.$$

Exercise 2.3

Solve the following recurrence relation. Determine all the constants. Express your solution with the simplest possible expression using the θ notation.

$$\begin{aligned} t_n &= 2t_{n-2} - t_{n-4} + 3n^2 + 2^n \qquad && n \geq 4 \\ t_n &= n && 0 \leq n \leq 3. \end{aligned}$$

Solution

There are 4 steps:

Step 1. General solution of the homogeneous equation $t_n = 2t_{n-2} - t_{n-4}$ without identifying the constants $\mapsto G(n)$

Step 2. Particular solution of the non homogeneous equation $t_n = 2t_{n-2} - t_{n-4} + 3n^2$ with identification of the constants $\mapsto p(n)$

Step 3. Particular solution of the non homogeneous equation $t_n = 2t_{n-2} - t_{n-4} + 2^n$ with identification of the constants $\mapsto q(n)$

Step 4. General solution of the non homogeneous equation $t_n = 2t_{n-2} - t_{n-4} + 3n^2 + 2^n$ with identification of the constants $\mapsto G(n) + p(n) + q(n)$

Let us now go through the details of each step.

Step 1. General solution of the homogeneous equation $t_n = 2t_{n-2} - t_{n-4}$.
Characteristic equation : $x^4 - 2x^2 + 1 = 0 \rightsquigarrow (x^2 - 1)^2$ because as the identity:

$$(a - b)^2 = a^2 + b^2 - 2ab.$$

Then $x^2 - 1$ is a difference of two squared terms, we use the identity

$$a^2 - b^2 = (a - b)(a + b).$$

It leads to: $x^2 - 1 = (x - 1)(x + 1)$.
Characteristic equation is then equivalent to:

$$(x - 1)^2(x + 1)^2 = 0.$$

Both two roots 1 and -1 are of multiplicity two.

$$G(n) = (C_1 + C_2 n)(1)^n + (C_3 + C_4 n)(-1)^n = C_1 + C_2 n + (C_3 + C_4 n)(-1)^n$$

$n = 2k$: $G(2k) = C_1 + C_3 + n(C_2 + C_4) = A_1 + B_1 n$

$n = 2k+1$: $G(2k+1) = C_1 - C_3 + n(C_2 - C_4) = A_2 + B_2 n$

It follows:

$$C_1 = \frac{A_1 + A_2}{2}, \quad C_2 = \frac{B_1 + B_2}{2}, \quad C_3 = \frac{A_1 - A_2}{2}, \quad C_4 = \frac{B_1 - B_2}{2}.$$

Step 2. Particular solution of the non homogeneous equation $t_n = 2t_{n-2} - t_{n-4} + 3n^2$
seek for $p(n) = An^4 + Bn^3 + Cn^2 + Dn + E$ degree 4
Why degree 4? The non homogeneous equation is of the form $t_n = 2t_{n-2} - t_{n-4} + f(n)$ where $f(n) = 3n^2$, i.e., $f(n)$ is of degree 2. So, the particular solution is a polynomial of degree at least 2. Since 1 is a root of the characteristic equation with multiplicity 2, it implies that the particular solution is a polynomial of degree 4.
$p(n)$ must satisfy

$$p(n) = 2p(n-2) - p(n-4) + 3n^2,$$

or equivalently

$$\begin{aligned} An^4 + Bn^3 + Cn^2 + Dn + E = {} & 2\Big(A(n-2)^4 + B(n-2)^3 + C(n-2)^2 + D(n-2) + E\Big) \\ & - \Big(A(n-4)^4 + B(n-4)^3 + C(n-4)^2 + D(n-4) + E\Big) + 3n^2. \end{aligned}$$

After developing the expression of the right hand side of the above equality, and after re-arranging the terms, we get:

n^4 terms	$A = 2A - A$
n^3 terms	$B = -16A + 2B + 16A - B$
n^2 terms	$C = 48A - 12B + 2C - 96A + 12B - C + 3$
	$48A = 3 \to A = 1/16$
n terms	$D = -64A + 24B - 8C + 2D + 256A - 48B + 8C - D$
	$0 = 192A - 24B \to B = \dfrac{192}{16 \times 24} = 1/2$
Constant terms	$E = 32A - 16B + 8C - 4D + 2E - 256A + 64B - 16C + 4D - E$
	$(32 - 256)A + (64 - 16)B - 8C = 0 \to C = 5/4$

$$p(n) = (1/16)n^4 + (1/2)n^3 + (5/4)n^2$$

Step 3. Particular solution of the non homogeneous equation $t_n = 2t_{n-2} - tn - 4 + 2^n$
$q(n) = C2^n$ must satisfy $q(n) = 2q(n-2) - q(n-4) + 2^n$
thus $C \times 2^n = 2C2^{n-2} - C2^{n-4} + 2^n \to \quad C = 16/9$

$$q(n) = (16/9)2^n$$

Step 4. General solution of the non homogeneous equation $t_n = 2t_{n-2} - tn - 4 + 3n^2 + 2^n$

$$t_n = G(n) + p(n) + q(n)$$

$n = 2k$

$$t_0 = 0 = A_1 + 16/9$$

$$t_2 = 2 = A_1 + 2B_1 + 2^4/16 + 2^3/2 + 5 \times 2^2/4 + 16 \times 2^2/9 = A_1 + 2B_1 + 1 + 4 + 5 + 2^6/9$$

$$2 = 2B_1 + 10 + 2^6/9 - 16/9 \quad \rightarrow \quad 2 = 2B_1 + (90 + 64 - 16)/9$$

$$\rightarrow \quad B_1 = \frac{18 - 138}{2 \times 9} = -120/18$$

$$A_1 = -16/9 \quad B_1 = -20/3$$

$$t_{2k} = -16/9 - 20n/3 + n^4/16 + n^3/2 + 5n^2/4 + 16 \times 2^n/9$$

$n = 2k + 1$

$$t_1 = 1 = A_2 + B_2 + 1/16 + 1/2 + 5/4 + 32/9$$

$$t_3 = A_2 + 3B_2 + (3^4)/16 + (3^3)/2 + (5 \times 9)/4 + (16/9) \times 2^3$$

$$A_2 = \frac{2011}{144} \quad B_2 = 55/3$$

$$t_{2k+1} = \frac{2011}{144} - (55/3)n + (5/4)n^2 + (1/2)n^3 + (1/16)n^4 + (16/9)2^n$$

$$t_n = \frac{195}{32} - 25n/2 + \left(-\frac{2267}{288} + 35n/6\right)(-1)^n + 5n^2/4 + n^3/2 + n^4/16 + (16/9) \times 2^n$$

$$t_n = \Theta(2^n)$$

Exercise 2.4

Solve the following recurrence relation

$$\begin{aligned} T(n) &= 5T(n/2) + (n \log n)^2 && n \geq 2 \\ T(n) &= 1 && n = 0, 1 \end{aligned}$$

where n is a power of 2. Express your answer, with the simplest possible expression using the θ notation.

Solution

Change of variable, we use $n = 2^k$

New variable: $S_k = T(2^k) = T(n)$.

New recurrence relation

$$S_k = 5S_{k-1} + (2^k \times k \log 2)^2 = 5S_{k-1} + 2^{2k} \times (\log 2)^2 \times k^2 = 5S_{k-1} + (\log 2)^2 k^2 4^k \tag{2.1}$$

Step 1: General solution for the homogeneous part. Characteristic equation is: $(x - 5) = 0$. General solution is then: $S_k^{(h)} = C_0 5^k$.

Step 2: Particular solution $S_k^{(p)}$ for the non homogeneous part.

- Look for a particular solution of the form:

  $$S_k^{(p)} = (C_1 + C_2 k + C_3 k^2) 4^k \tag{2.2}$$

  See slide 28 in Lecture 1.

- How to identify the constants in (2.2)? We embed $S_k^{(p)}$ in the recurrence relation (2.1), we get

  $$(C_1 + C_2 k + C_3 k^2) 4^k = 5 \times \left(\left[C_1 + C_2(k-1) + C_3(k-1)^2 \right] 4^{k-1} \right) + (\log 2)^2 k^2 4^k.$$

  Let us divide by 4^{k-1}, we get

  $$4(C_1 + C_2 k + C_3 k^2) = 5 \times \left(C_1 + C_2(k-1) + C_3(k-1)^2 \right) + 4(\log 2)^2 k^2.$$

  Since this last relation must hold for all k, we deduce the following relations:

Constant terms	$4C_1 = 5(C_1 - C_2 + C_3)$	$\rightarrow C_1 - 5C_2 + 5C_3 = 0$
Linear Terms w.r.t k	$4C_2 = 5(C_2 - 2C_3)$	$\rightarrow C_2 - 10C_3 = 0$
Quadratic terms w.r.t. k^2	$4C_3 = 5C_3 + 4(\log 2)^2$	$\rightarrow C_3 = -4(\log 2)^2$.

It follows:

$$C_1 = -180 \times (\log 2)^2$$
$$C_2 = -40 \times (\log 2)^2$$
$$C_3 = -4 \times (\log 2)^2.$$

Step 3: General solution for the non homogeneous part.

$$S_k = S_k^{(p)} + S_k^{(h)} = C_0 5^k - (\log 2)^2 (180 + 40k + 4k^2) 4^k.$$

Step 4a: General solution for the non homogeneous part with respect to the S_k recurrence relation. We need to compute C_0. Its value can be deduced using the initial conditions of the recurrence relations.
We use the initial condition $T(1) = 1$ in order to identify the C_0 constant. Indeed, C_0 must satisfy:

$$T(1) = S_0 = 1 = C_0 - 180(\log 2)^2 \quad \rightarrow \quad C_0 = 1 + 180(\log 2)^2.$$

Therefore, $S_k = (1 + 180(\log 2)^2) 5^k - (\log 2)^2 (180 + 40k + 4k^2) 4^k$.

Step 4b: General solution for the non homogeneous part with respect to the $T(n)$ recurrence relation.

$$T(n) = T(2^k) = S_k = (1 + 180(\log 2)^2) 5^{\log_2 n} - (\log 2)^2 (180 + 40 \log_2 n + 4(\log_2 n)^2) 4^{\log_2 n}$$

i.e.,

$$T(n) = (1 + 180(\log 2)^2) n^{\log_2 5} - (\log 2)^2 (180 + 40 \log_2 n + 4(\log_2 n)^2) n^{\log_2 4}.$$

Therefore,

$$T(n) = \Theta(n^{\log_2 5}).$$

Exercise 2.5 *(Townsend [14], Exercise 29, p. 144)*

Find a recurrence relation for the number of ways to pair off $2n$ people for n tennis matches. For example, if $n = 2$, then there are three pairings:

- $\{1, 2\}$ *and* $\{3, 4\}$,
- $\{1, 3\}$ *and* $\{2, 4\}$,
- $\{1, 4\}$ *and* $\{2, 3\}$.

Solution

Let $S(n)$ number of ways to pair off $2n$ people. Consider $2n + 2$ people. Choose a particular person A from the people. There are $2n + 1$ possible people to pair with A. Suppose that A pairs off with a person B, thus there are $S(n)$ ways to pair off the rest. Thereby, we have the following equation:

$$S(n + 1) = (2n + 1)S(n).$$

Exercise 2.6 *(Townsend [14], Exercise 7a, p. 161)*

Solve the following recurrence relation

$$a_0 = 1, \qquad a_n + 2a_{n-1} = 2^n - n^2 \quad (n \geq 1).$$

Solution

The solution of this recurrence relation is as follows,

$$a_n = f(n) + g(n) + h(n),$$

where:

- $f(n)$ is the solution of $f(n) + 2f(n-1) = 0$
- $g(n)$ is a particular solution of $g(n) + 2g(n-1) = 2^n$
- $h(n)$ is a particular solution of $h(n) + 2h(n-1) = -n^2$.

Solving $f(n)+2f(n-1) = 0$ can be done using the characteristic equation: $x+2 = 0 \Leftrightarrow x = -2$. We then have

$$f(n) = c(-2)^n,$$

where c is an arbitrary constant.

Guessing $g(n) = a2^n$. Replacing $g(n)$ by this particular solution in $g(n) + 2g(n-1) = 2^n$, we have

$$a2^n + 2a2^{n-1} = 2^n \qquad \Leftrightarrow \qquad a = \frac{1}{2}.$$

We deduce: $g(n) = 2^{n-1}$.

Guessing $h(n) = a_2n^2 + a_1n + a_0$. Replacing $h(n)$ in relation $h(n) + 2h(n-1) = -n^2$, we have

$$\begin{aligned}
& a_2n^2 + a_1n + a_0 + 2(a_2(n-1)^2 + a_1(n-1) + a_0) = -n^2 \\
\Leftrightarrow \quad & n^2(3a_2) + n(3a_1 - 4a_2) + 3a_0 - 2a_1 + 2a_2 = -n^2 \\
\Leftrightarrow \quad & \begin{cases} 3a_2 & = -1 \\ 3a_1 - 4a_2 & = 0 \\ 3a_0 - 2a_1 + 2a_2 & = 0 \end{cases} \\
\Leftrightarrow \quad & \begin{cases} a_2 & = -\frac{1}{3} \\ a_1 & = -\frac{4}{9} \\ a_0 & = -\frac{2}{27} \end{cases}
\end{aligned}$$

From these analysis, we have

$$a_n = c(-2)^n + 2^{n-1} - \frac{1}{3}n^2 - \frac{4}{9}n - \frac{2}{27} \tag{2.3}$$

Using the initial condition $a_0 = 1$ we have:

$$1 = c + 2^{-1} - \frac{2}{27} \tag{2.4}$$

$$\Leftrightarrow c = \frac{2}{27} + 1 - \frac{1}{2} = \frac{31}{54} \tag{2.5}$$

Finally, the solution of the given system is:

$$a_n = \frac{31}{54}(-2)^n + 2^{n-1} - \frac{1}{3}n^2 - \frac{4}{9}n - \frac{2}{27} \tag{2.6}$$

Exercise 2.7

Find asymptotic lower/upper bounds for $T(n)$. Assume $T(n)$ is constant for small n.

(a) $T(n) = 3T(n/2) + \sqrt{n}$

(b) $T(n) = 5T(n/5) + n \log n$

(c) $T(n) = \sqrt{n}\, T(\sqrt{n}) + n$

(d) $T(n) = T(n-2) + 2 \log n$

Solution

(a) Apply the Master theorem with $a = 3$ and $b = 2$. We compare $\sqrt{n} = n^{\frac{1}{2}}$ with $n^{\log_2 3}$. We deduce:

$$n^{\frac{1}{2}} = n^{\log_2 3 - (\log_2 3 - \frac{1}{2})}.$$

But $\log_2 3 > \frac{1}{2}$, so we apply Case 1 of the Master theorem:

$$T(n) = \Theta(n^{\log_2 3}).$$

(b) Solution 1

We cannot directly apply the Master theorem, see comments on page 75 of the textbook [3] (p. 95 in the 3rd edition [4]). The reason is as follows. A priori, the recurrence relation has the proper form: $a = 5$, $b = 5$, $f(n) = n \log n$. Let us evaluate in which cases, we fall. We therefore evaluate $n^{\log_b a}$. We have

$$n^{\log_b a} = n^{\log_5 5} = n.$$

We cannot therefore be in Cases 1 or 2. For Case 3, let us check whether the regularity condition holds for $f(n)$.

$$\frac{f(n)}{n^{\log_b a}} = \frac{n \log n}{n^{\log_5 5}} = \log n.$$

It means that we do not have $f(n)\Omega(n^{\log_b a + \varepsilon})$, and therefore we cannot apply the Master theorem.

However, we can apply the result of exercise 4.4-2, p. 84 in [3] (or exercise 4.6-2 p. 106 in [4]), with $a = 5$ and $b = 5$. The result is as follows:
If $f(n) = \Theta(n^{\log_b a \log^k n})$, where $k \geq 0$,
then the master recurrence has solution $T(n) = \Theta(n^{\log_b a} \log^{k+1} n)$.

Note that: $\log^k n = (\log n)^k$.

Let us compute $n^{\log_b a \log^k n}$ with $a = 5$ and $b = 5$:

$$n^{\log_b a \log^k n} = n \log^k n.$$

Therefore $f(n) = n^{\log_b a \log^k n}$ for $k = 1$.

We then obtain:

$$T(n) = \Theta(n \log^2 n).$$

(b) Solution 2

Alternate solution for getting bounds. Use that:

$$5T(\frac{n}{5}) \leq T(n) = 5T(\frac{n}{5}) + n \log n \leq 5T(\frac{n}{5}) + n^2,$$

and solve the two following recurrence equations:

$$5T(\frac{n}{5}) = T(n) \qquad \text{and} \qquad T(n) = 5T(\frac{n}{5}) + n^2.$$

Solving $5T(\frac{n}{5}) = T(n)$ is easy. Take $n = 5^k$ to reduce to the following recurrence relation w.r.t. k: $5a_k = a_{k-1}$, where $a_k = T(n)$. We deduce that: $a_k = 5^k$, i.e., $T(n) = n$.

If the two recurrence relations have the same solution, then we will get a $\Theta()$ complexity, otherwise (that will be the case here) we get lower and upper bounds on the time complexity.

Solving the first recurrence relation leads to: $T(n) \geq \Theta(n)$.

Solving the second recurrence relation leads to: $T(n) \leq \Theta(n^2)$ (use the result on Slide 33 of Lecture 1 with $a = b = 5$ and $k = 2$).

(c)

$$\begin{aligned} T(n) &= n^{\frac{1}{2}} T(n^{\frac{1}{2}}) + n = n^{\frac{1}{2}} \left(n^{\frac{1}{4}} T(n^{\frac{1}{4}}) + n \right) + n = n^{\frac{1}{2}} n^{\frac{1}{4}} \left(n^{\frac{1}{8}} T(n^{\frac{1}{8}}) + n \right) + 2n \\ &= \cdots = n^{\frac{1}{2} + \frac{1}{4} + \ldots + \frac{1}{2^k}} T(n^{\frac{1}{2^k}}) + kn = n \left[k + n^{\frac{1}{2^{k+1}}} T(n^{\frac{1}{2^k}}) \right] \end{aligned}$$

where k is an arbitrary number.

In order to simplify the last expression, let $k = \log_2(\log_2 n)$ or $n = 2^{2^k}$, then we have:

$$T(n) = n \left[\log_2(\log_2(n)) + \frac{1}{2} T(2) \right].$$

We conclude: $T(n) = \Theta(n \log_2(\log_2 n)) = \Theta(n \log(\log n))$.

Keeping $\log_2 n$ instead of $\log n$ is a priori not necessary. Remember that we are interested in asymptotic behavior.

(d)

$$
\begin{aligned}
T(n) &= T(n-2) + 2\log n \\
&= T(n-4) + 2\log(n-2) + 2\log n \\
&= T(n-6) + 2\log(n-4) + 2\log(n-2) + 2\log n \\
&= \ldots \\
&= T(n-2k) + 2\left[\log(n-2(k-1)) + \ldots + \log n\right]
\end{aligned}
$$

Assume first that n is even. Let $k = \frac{n}{2}$, we have:

$$T(n) = T(0) + 2\left[\log 2 + \log 4 + \cdots + \log n\right].$$

It is easy to see that:

$$T(n) > T(0) + \log 1 + \log 2 + \ldots + \log n \quad \text{and} \quad T(n) < T(0) + 2\left[\log 1 + \log 2 + \ldots + \log n\right].$$

Similar computation when n is odd.

In conclusion, we have:

$$T(n) = \Theta\left(\log 1 + \log 2 + \ldots + \log n\right) = \Theta(\log(n!)).$$

Exercise 2.8

We propose to analyze the complexity of a generalization of the mergesort algorithm. In the regular mergesort algorithm, we break the list into two equal size sublists, we run the mergesort recursively on each sublist and then merge the 2 sorted sublists. Consider the following variation of mergesort, called k-ary-Mergesort: We break the list into k sublists of equal size (for simplification of analysis assume that $n = k^t$ for some $t \geq 0$), we run the k-ary-Mergesort recursively on each sublist and then we merge the k sorted lists.

(a) *Describe an algorithm in order to partition the initial list into k sublists of size $\frac{n}{k}$. Perform the complexity analysis of your algorithm, and express the complexity using the input-size parameters n and k.*

(b) *Design a k-merge algorithm for k-ary-Mergesort - describe k-merge in clear English - so that k sorted lists each of size $\frac{n}{k}$ can be merged into one sorted list. Perform the complexity analysis of your algorithm, and express the complexity using the input-size parameters n and k. Indeed, show that the number of operations satisfies the following recurrence relation:*

$$T_{\text{MERGE}} = O(k) + T_{\text{MERGE}}(n-1, k).$$

(c) *Describe in clear English the k-ary-mergesort algorithm (that calls your k-Merge), and write the recurrence for the running time on $n = k^t$ items. Solve it to find the order of growth of the runtime of k-ary-Mergesort. Discuss how it compares to regular mergesort.*

(d) *Provide the number of operations which is required by the k-ary-Mergesort for $k = 2$ and $k = 3$.*

(e) *What is the value of k for which the complexity is minimum?*

a. Let $B[i][1], B[i][2] \ldots B[i][\frac{n}{k}] \ldots$ the i^{th} sublist that is created from the input list $A[1 \ldots n]$. The algorihm to partition the initial list into k sublists can be described as following:

Procedure PARTITION(A, n, k)

$u = 1$
$v = 0$
for $i = 1 \ldots n$ **do**
  $v = v + 1$
  $B[u][v] = A[i]$
  if $v = \frac{n}{k}$ **then**
    $u = u + 1$
    $v = 0$
  end if

end for
return B

End Procedure

Because, we have to move all elements in A to the corresponding $B[u][v]$ by a for statement. Thus, we can see easily that the complexity of $PARTITION$ is $\Theta(n)$.

b. In order to merge sublists $B[1], B[2] \dots B[k]$, we first find the smallest element among the set made of the first elements of these sublists. Then, we remove this smallest element from the corresponding sublist, and put this smallest element to the result list A. We go on by recursively calling the MERGE algorithm. This process is repeated until all sublists are empty. The MERGE algorithm can be described as follows:

Procedure MERGE(A, sublists $B[1], B[2] \dots B[k], n, k$)
if $n = 0$ **then**
  return A
end if
Let $b_1, b_2 \dots b_k$ the first elements of $B[1], B[2] \dots B[k]$, respectively.
Find $b_{\min} = \arg\min\{b_1, b_2, \dots, b_k\}$
Remove $b_{\min}$ from $B[\min]$
Add $b_{\min}$ to A
MERGE(A, sublists $B[1], B[2] \dots B[k], n-1, k$)
return A

End Procedure

Comment. The MERGE procedure could have also been written as a non recursive algorithm.

Let $T_{\text{MERGE}}(n, k)$ is number of operations of the MERGE algorithm. The MERGE algorithm takes k operations to select the smallest element from all sublists and takes $T_{\text{MERGE}}(n-1, k)$ for the recursive call with $n-1$ elements. We therefore deduce the following recurrence relation:

$$\begin{aligned}
T_{\text{MERGE}}(n, k) &= k + T_{\text{MERGE}}(n-1, k) \\
&= 2k + T_{\text{MERGE}}(n-2, k) \\
&= 3k + T_{\text{MERGE}}(n-3, k) \\
&= \dots \\
&= nk + T_{\text{MERGE}}(0, k)
\end{aligned}$$

We conclude that: $T_{\text{MERGE}}(n, k) = \Theta(nk)$.

c. The k-ary-mergesort uses PARTITION algorithm to partition input list A into k sublists $B[1], B[2] \ldots B[k]$. Then, k-ary-mergesort is used for each sublist $B[i]$. Finally, the MERGE algorithm is called to merge sorted sublists $B[1], B[2] \ldots B[k]$. k-ary-mergesort can be described as following:

Procedure MERGESORT(A , n , k)

if $n = 1$ **then**
  return A
end if

$B[1], B[2] \ldots B[k] =$ PARTITION(A, n, k)

for $i = 1 \ldots k$ **do**
  MERGESORT($B[i], \frac{n}{k}, k$)
end for

MERGE(A, sublists $B[1], B[2] \ldots B[k], n, k$)

return A

End Procedure

Let $T_{\text{MERGESORT}}(n, k)$ number of operations of MERGESORT. Because MERGESORT takes $\Theta(n)$ operations to partition A into k sublists, takes $T_{\text{MERGESORT}}(\frac{n}{k}, k)$ operations for each sublist and takes $T_{MERGE}(n, k)$ to merge sorted sublists. Thus, we have the following relation:

$$\begin{aligned}
T_{\text{MERGESORT}}(n, k) &= kT_{\text{MERGESORT}}(\frac{n}{k}, k) + \Theta(n) + T_{\text{MERGE}}(n, k) \\
&= kT_{\text{MERGESORT}}(\frac{n}{k}, k) + \Theta(n) + \Theta(nk) \\
&= kT_{\text{MERGESORT}}(\frac{n}{k}, k) + \Theta(nk) \\
&= k^2T_{\text{MERGESORT}}(\frac{n}{k^2}, k) + \Theta(nk) + \Theta(nk) \\
&= k^3T_{\text{MERGESORT}}(\frac{n}{k^3}, k) + \Theta(nk) + \Theta(nk) + \Theta(nk) \\
&= \ldots
\end{aligned}$$

Suppose that $n = k^t$ or $t = \log_k(n)$, we have

$$\begin{aligned}
T_{\text{MERGESORT}}(n, k) &= k^tT_{\text{MERGESORT}}(\frac{n}{k^t}, k) + t\Theta(nk) \\
&= nT_{\text{MERGESORT}}(1, k) + \log_k(n)\Theta(nk) \\
&= \Theta(\log_k(n)nk)
\end{aligned}$$

The complexity of k-ary-mergesort is $\Theta(\log_k(n)nk) = \Theta(\log(n)n\frac{k}{\log(k)})$. We know that complexity of the conventional merge sort ($k = 2$) is $\Theta(\log(n)n)$. Thus, $\frac{k}{\log(k)}$ decides the performance of k-ary-mergesort over the conventional one.

d.

For $k = 2$, we have

$$T_{MERGESORT}(n, 2) = \Theta(\log(n)n2) = \Theta(\log(n)n) \tag{2.7}$$

For $k = 3$, we have

$$T_{MERGESORT}(n, 3) = \Theta(\log_3(n)n3) = \Theta(\log_3(n)n) \tag{2.8}$$

e. In general, we have $T_{MERGESORT}(n, k) = \Theta(\log_k(n)nk) = \Theta(\log(n)n\frac{k}{\log(k)})$. In order to mimimize complexity, we have to mimimize $\frac{k}{\log(k)}$.

$$\frac{d}{dk}\left[\frac{k}{\log(k)}\right] = 0 \tag{2.9}$$

$$\Leftrightarrow \frac{1}{\log(k)} - \frac{1}{\log^2(k)\ln(2)} = 0 \tag{2.10}$$

$$\Leftrightarrow \log(k)\ln(2) = 1 \tag{2.11}$$

$$\Leftrightarrow \ln(k) = 1 \tag{2.12}$$

$$\Leftrightarrow k = e = 2.71828183 \tag{2.13}$$

So, k about 2 or 3 is the optimal value for k-ary-mergesort.

Exercise 2.9 *(Townsend [14], Exercise 6, p. 175)*

Consider the problem of multiplying together two $n \times n$ matrices. Each addition and multiplication counts as one operation.

(a) *Show that the usual multiplication algorithm requires roughly n^3 operations*

(b) *Show that two $n \times n$ matrices can be added using n^2 additions.*

(c) *Let*

$$A = \begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix} \quad \text{and} \quad B = \begin{pmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{pmatrix}$$

be two $n \times n$ matrices, where the A_{ij} and B_{ij} are $n/2 \times n/2$ matrices. Show that

$$AB = \begin{pmatrix} C_{11} & C_{12} \\ C_{21} & C_{22} \end{pmatrix}$$

where

$$\begin{aligned} C_{11} &= A_{11}B_{11} + A_{12}B_{21} \\ C_{12} &= A_{11}B_{12} + A_{12}B_{22} \\ C_{21} &= A_{21}B_{11} + A_{22}B_{21} \\ C_{22} &= A_{21}B_{12} + A_{22}B_{22}. \end{aligned}$$

(d) *Using (b) and (c) show that if x multiplications and y additions are required to multiply two 2×2 matrices, then the number of operations required to multiply two $n \times n$ matrices (assuming n is a power of two) satisfies $a_1 = 1, a_n = xa_{n/2} + yn^2/4$.*

(e) *Solve the recurrence relation obtained in (d).*

(f) *Strassen (1969) provided a technique in which $x = 7$ and $y = 18$. Using these values, how does the Strassen algorithm compare to the usual technique?*

a. Let C the matrix that is created by multiplication of A and B. We have

$$c_{ij} = \sum_{k=1\ldots n} a_{ik}b_{kj} \tag{2.14}$$

Evidently, for computing c_{ij}, it takes n multiplications and $n-1$ additions. The matrix C has $n \times n$ element c_{ij}. Thus, for doing multiplication of A and B, we use n^3 multiplications and $n^2(n-1)$ additions.

b. If $C = A + B$ then we have $c_{ij} = a_{ij} + b_{ij}$. For computing c_{ij}, we use one addition, hence for computing C which has $n \times n$ elements, we use $1 \times n \times n = n^2$ additions.

c. Let $c_{ij} \in C_{11}$. So we have $i, j \leq \frac{n}{2}$.

c_{ij} is computed as

$$c_{ij} = \sum_{k=1\ldots n} a_{ik} b_{kj} \tag{2.15}$$

$$= \sum_{k=1\ldots\frac{n}{2}} a_{ik} b_{kj} + \sum_{k=\frac{n}{2}+1\ldots n} a_{ik} b_{kj} \tag{2.16}$$

If $k \leq \frac{n}{2}$ then $a_{ik} \in A_{11}$ and $b_{kj} \in B_{11}$. If $k > \frac{n}{2}$ then $a_{ik} \in A_{12}$ and $b_{kj} \in B_{21}$. Thus, we know $\sum_{k=1\ldots\frac{n}{2}} a_{ik} b_{kj}$ represents multiplication of row i of A_{11} and col j of B_{12}. $\sum_{k=\frac{n}{2}+1\ldots n} a_{ik} b_{kj}$ represents multiplication of row i of A_{12} and col j of B_{21}. Therefore, we have

$$C_{11} = A_{11} B_{11} + A_{12} B_{21} \tag{2.17}$$

Similarly for C_{12}, C_{21} and C_{22}.

d. Let a_n number of operation to multiply two matrixes $n \times n$ A and B. Dividing A into four $\frac{n}{2} \times \frac{n}{2}$ sub-matrixes. Similarly for B, we have:

$$A = \begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix} \tag{2.18}$$

$$B = \begin{pmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{pmatrix} \tag{2.19}$$

Consider each submatrix as a number, the multiplication of A and B becomes multiplication of two 2×2 matrixes so it takes x multiplications and y additions. However, we do actually multiplication of $\frac{n}{2} \times \frac{n}{2}$ submatrixes instead of multiplication of two numbers. This action takes $xa_{\frac{n}{2}}$ operations. Similarly, we do actually additions of $\frac{n}{2} \times \frac{n}{2}$ submatrixes instead of additions of two numbers. This actions takes $y[\frac{n}{2}]^2$.
Totally, $A \times B$ takes $xa_{\frac{n}{2}} + y\frac{n^2}{4}$ operations. We have following relation:

$$a_n = xa_{\frac{n}{2}} + y\frac{n^2}{4}.$$

e. Let $n = 2^t$, we have:

$$a_{2^t} = xa_{2^{t-1}} + y4^{t-1}.$$

Let $u_t = a_{2^t}$, we have:

$$u_t = xu_{t-1} + y4^{t-1}.$$

This is equivalent to:

$$u_t - xu_{t-1} = y4^{t-1}.$$

Let us solve this last recurrence relation. For the homogeneous part,

$$u_t - xu_{t-1} = 0.$$

cx^t is a solution of it.

A particular solution of

$$u_t - xu_{t-1} = y4^{t-1}$$

is $u_t = h4^t$ if $4 \neq x$ (otherwise, the solution is $u_t = ht4^t$, and the solution is left to the reader in that particular case). We have:

$$\begin{aligned} h4^t - xh4^{t-1} &= y4^{t-1} \\ 4h - xh &= y \\ h &= -\frac{y}{x-4}. \end{aligned}$$

So, $u_t = cx^t - \frac{y}{x-4}4^t = cx^{\log_2 n} - \frac{y}{x-4}n^2$, i.e., $a_n = cx^{\log_2 n} - \dfrac{y}{x-4}n^2$.

Remember that $a_1 = 1$. Therefore, $cx^{\log_2 1} - \frac{y}{x-4}1^2 = 1$, i.e. $c - \dfrac{y}{x-4} = 1$.

It follows: $c = 1 + \dfrac{y}{x-4}$.

Thus, $a_n = \left[1 + \dfrac{y}{x-4}\right] x^{\log_2 n} - \dfrac{y}{x-4}n^2 = \left[1 + \dfrac{y}{x-4}\right] n^{\log_2 x} - \dfrac{y}{x-4}n^2.$

f. With $x = 7$ and $y = 18$ we have

$$a_n = 7^{1+\log_2 n} - 6n^2 = 7n^{\log_2 7} - 6n^2 < 7n^3 - 6n^2.$$

Conventional method takes $x = 8$ multiplications and $y = 4$ additions. In such a case, we have:

$$a_n = 2.8^{\log(n)} - n^2 = 2n^3 - n^2.$$

As $\log_2(7) < \log_2(8) = 3$. Thus $7n^{\log_2(7)} - 6n^2$ has a lower increase rate than $2n^3 - n^2$. That is why Strassen's technique is faster than the conventional one.

Note that y does not play a role in the value of the order of growth, but only in the hidden constant. In other words, y does not impact the asymptotic behavior, but does impact the practical complexity in terms of computing times: It cannot be arbitrarily large without impact on the computing times, even if it does not impact the asymptotic behavior.

Exercise 2.10

Solve the following recurrence relation:

$$a_n = 2(a_{n-1} - a_{n-2}), \qquad n \geq 0, \qquad a_0 = 1, a_1 = 2.$$

You will need to remember the following results:

- *De Moivre's Theorem:* $(\cos\theta + i\sin\theta)^n = \cos n\theta + i \sin n\theta, \qquad n \geq 0$
- *If* $z = x + iy \in \mathbb{C}, z \neq 0$, *then*

$$z = \sqrt{x^2+y^2}\left(\frac{x}{\sqrt{x^2+y^2}} + i\frac{y}{\sqrt{x^2+y^2}}\right) = r(\cos\theta + i\sin\theta)$$

$z^n = r^n(\cos n\theta + i \sin n\theta)$

1. Set the characteristic equation: $r^2 - 2r + 2 = 0$

2. Solve the characteristic equation. Two roots: $r_1 = 1+i, r_2 = 1-i$.
Trigonometric form of the two roots: $r_1 = \sqrt{2}\left(\cos\frac{\pi}{4} + i\sin\frac{\pi}{4}\right)$, $r_2 = \sqrt{2}\left(\cos\frac{\pi}{4} - i\sin\frac{\pi}{4}\right)$.

3. General form of the solution:

$$\begin{aligned}
a_n &= C_1(1+i)^n + C_2(1-i)^n \\
&= C_1\left[\sqrt{2}\left(\cos\frac{\pi}{4} + i\sin\frac{\pi}{4}\right)\right]^n + C_2\left[r_2 = \sqrt{2}\left(\cos\frac{\pi}{4} - i\sin\frac{\pi}{4}\right)\right]^n \\
&= C_1(\sqrt{2})^n\left(\cos\frac{n\pi}{4} + i\sin\frac{n\pi}{4}\right) + C_2(\sqrt{2})^n\left(\cos\frac{n\pi}{4} - i\sin\frac{n\pi}{4}\right) \\
&= (\sqrt{2})^n\left[(C_1+C_2)\cos\frac{n\pi}{4} + i(C_1-C_2)\sin\frac{n\pi}{4}\right]
\end{aligned}$$

4. Determine the values of the constants, using the initial conditions.
$a_0 = 1 = C_1 + C_2$
$a_1 = 2 = (C_1 + C_2) + (C_1 - C_2)i \quad \rightarrow \quad C_1 - C_2 = -i$
It follows:

$$a_n = (\sqrt{2})^n\left(\cos\frac{n\pi}{4} + \sin\frac{n\pi}{4}\right).$$

Exercise 2.11 *(Manber, exercise 3.26 p. 59 - MidTerm - Fall 2010)*

Find the asymptotic behavior of the function $T(n)$ defined by the recurrence relation:

$$T(n) = T(n/2) + \sqrt{n} \qquad T(1) = 1.$$

You can consider only values of n that are powers of 2.

Solution

Preliminary step: Reduction to a linear recurrence relation

Change of variables: $n = 2^k$. The recurrence relation becomes:

$$T(2^k) = T(2^{k-1}) + \sqrt{2^k}.$$

Let $A(k) = T(2^k)$. Then, $A(0) = T(1) = 1$, and

$$A(k) = A(k-1) + 2^{k/2}.$$

The solution of this last recurrence relation goes as follows.

Step 1 Generic solution of the homogeneous linear part of the equation.

1. Set the characteristic equation: $r - 1 = 0$

2. Solve the characteristic equation. One unique root: $r_1 = 1$.

3. General form of the solution: $A(k) = C_1 \times 1^k = C_1$ where C_1 is a constant to be determine in Step 3.

Step 2 Particular solution of the non homogeneous linear part of the equation.

The general form of such a particular solution is:

$$C_2 \times 2^{k/2},$$

and must satisfy

$$C_2 \times 2^{k/2} = C_2 \times 2^{(k-1)/2} + 2^{k/2}.$$

It follows that:

$$\sqrt{2} \times C_2 = C_2 + \sqrt{2} \qquad \hookrightarrow \qquad C_2 = \frac{\sqrt{2}}{\sqrt{2} - 1}.$$

Step 3 Solution of the non homogeneous linear part of the equation. Index k

From the two previous steps, it follows that:

$$A(k) = C_1 + \frac{\sqrt{2}}{\sqrt{2}-1} 2^{k/2}.$$

But $A(0) = 1 = C_1 + \frac{\sqrt{2}}{\sqrt{2}-1}$. It follows that:

$$C_1 = \frac{1}{1-\sqrt{2}}.$$

Step 4 Solution of the non homogeneous linear part of the equation. Index n

$$T(n) = \frac{1}{1-\sqrt{2}} + \frac{\sqrt{2}}{\sqrt{2}-1} 2^{\frac{\log_2 n}{2}} = \frac{1}{1-\sqrt{2}} + \frac{\sqrt{2}}{\sqrt{2}-1} \sqrt{n}.$$

It follows:

$$T(n) = O(\sqrt{n}).$$

Note that:

$$\frac{1}{1-\sqrt{2}} = \frac{1+\sqrt{2}}{1-2} = -(1+\sqrt{2}).$$

$$\frac{\sqrt{2}}{\sqrt{2}-1} = \frac{\sqrt{2}(\sqrt{2}+1)}{1} = 2+\sqrt{2}.$$

It leads to:

$$T(n) = (\sqrt{2}+1)(\sqrt{2n}-1).$$

Exercise 2.12

Solve the following recurrence relations using the characteristic equation:

(i) $t_n = 6t_{n-1} - 9t_{n-2} + (n^2 - 5n)7^n$ *for* $n > 1$, $t_0 = 0, t_1 = 1$.

(ii) $t_n = 2t_{\frac{n}{5}} + 6n^3$ *for* $n > 1$, n *a power of 5* , $t_1 = 6$.

Solution of Recurrence Relation *(i)*

The associated homogeneous recurrence relation is:

$$a_n = 6a_{n-1} - 9a_{n-2} \rightarrow r^2 - 6r + 9 = (r-3)^2$$

So we have

$$a_n^h = (c_1 + c_2 n) \times 3^n.$$

The non-homogeneous part is:

$$a_n^p = p(n)\beta^n = (b_0 n^2 + b_1 n + b_2)7^n \rightarrow b_0 = \frac{49}{16}, b_1 = -\frac{49}{2}, b_2 = \frac{4851}{128}.$$

The solution is:

$$a_n = a_n^h + a_n^p = (c_1 + c_2 n)3^n + (\frac{49}{16}n^2 - \frac{49}{2}n + \frac{4851}{128})7^n.$$

Using the initial conditions, we have

$$a_n = a_n^h + a_n^p = (-\frac{4851}{128} - \frac{17}{96}n)3^n + (\frac{49}{16}n^2 - \frac{49}{2}n + \frac{4851}{128})7^n.$$

Solution of Recurrence Relation *(ii)*

We assume that $n = 5^k$. Then we have $\log_5 n = k$. So

$$t_k = 2t_{k-1} + 6 \times 5^{3k}$$

The associated homogeneous part is:

$$t_k = 2t_{k-1} \rightarrow t_k^h = c_1 2^k$$

The non-homogeneous part is:

$$t_k^p = b_1 125^k \to b_1 125^k = 2b_1(125)^{k-1} + 6 \times 125^k \to b_1 = \frac{250}{41}$$

The solution is:

$$t_k = t_k^h + t_k^p = c_1 2^k + \frac{250}{41} 125^k \to c_1 = -\frac{4}{41}.$$

Thus,

$$t_k = -\frac{4}{41} 2^k + \frac{250}{41} 125^k.$$

Since $k = \log_5 n$, therefore:

$$t_n = -\frac{4}{41} 2^{\log_5 n} + \frac{250}{41} 125^{\log_5 n} = -\frac{4}{41} n^{\log_5 2} + \frac{250}{41} n^3.$$

Exercise 2.13

Solve the following system of recurrence relations:

$$a_0 = 1 \qquad b_0 = c_0 = 0 \tag{2.20}$$
$$a_n = 2a_{n-1} + b_{n-2} + c_{n-1} \qquad n \geq 1 \tag{2.21}$$
$$b_n = b_{n-1} - c_{n-1} + 4^{n-1} \qquad n \geq 1 \tag{2.22}$$
$$c_n = c_{n-1} - b_{n-1} + 4^{n-1} \qquad n \geq 1. \tag{2.23}$$

Solution

Add (2.21) and (2.22) leads to:

$$b_n + c_n = 2 \times 4^{n-1} \quad \text{for } n \geq 1.$$

This last recurrence relation can be rewritten:

$$c_{n-1} = 2 \times 4^{n-2} - b_{n-1} \quad \text{for } n \geq 2. \tag{2.24}$$

Replace c_{n-1} in (2.22). It leads to:

$$b_n = b_{n-1} - 2 \times 4^{n-2} + b_{n-1} + 4^{n-1} = 2 \times b_{n-1} + 2 \times 4^{n-2}.$$

Characteristic equation of the homogeneous recurrence relation is: $r - 2 = 0$.

Result is: $b_n = C_1 2^n$.

Search for a particular solution under the form $C_2 4^n$

$$C_2 4^n - 2C_2 4^{n-1} = 2 \times 4^{n-2} \qquad \rightsquigarrow \qquad C_2 = \frac{1}{4}.$$

We deduce:

$$b_n = C_1 2^n + 4^{n-1} \qquad \text{for } n \geq 2. \tag{2.25}$$

Using (2.22) and (2.23) as well as the initial conditions, we get $b_1 = c_1 = 1$, and then $b_2 = 4$.

Writing (2.25) for $n = 2$ leads to $C_1 = 0$.

It follows: $b_n = 4^{n-1}$ for $n \geq 1$, as $b_1 = 1 = 4^{1-1}$.

According to (2.25), we deduce $c_n = 4^{n-1}$ for $n \geq 1$.

It follows:
$$b_{n-1} = c_{n-1} = 4^{n-2} \qquad \text{for} \quad n \geq 2. \tag{2.26}$$

Using (2.24) and (2.26) in order to eliminate b_{n-2} and c_{n-1} in (2.21) leads to:

$$a_n - 2a_{n-1} = 2 \times 4^{n-2} \qquad n \geq 2.$$

We observe that it is the same recurrence relation as for b_n, consequently $a_n = C_1 \times 2^n + 4^{n-1}$.

Compute a_2. We get $a_2 = 6$, hence $C_1 = \frac{1}{2}$.

We deduce: $a_n = 2^{n-1} + 4^{n-1} \qquad n \geq 2$.

This last relation is also valid for $n = 1$ as $a_1 = 2 = 2^{1-1} + 4^{1-1}$.

Hence, $a_n = 2^{n-1} + 4^{n-1} \qquad n \geq 1$.

Summary:
$$a_n = 2^{n-1} + 4^{n-1} \quad \text{and} \quad b_n = c_n = 4^{n-1} \qquad n \geq 1.$$

Exercise 2.14

Solve the system of recurrence relations

$$\begin{cases} s_n = 8s_{n-1} - 9t_{n-1} \\ t_n = 6s_{n-1} - 7t_{n-1}. \end{cases}$$

Initial conditions: $s_0 = 4, t_0 = 1$.

Solution

Subtracting the second recurrence relation to the first recurrence relation leads to:

$$s_n - t_n = 2(s_{n-1} - t_{n-1}). \tag{2.27}$$

Set $a_n = s_n - t_n$. Recurrence relation (2.27) becomes:

$$a_n = 2a_{n-1} \tag{2.28}$$

This is a linear and homogeneous recurrence relation, such that its characteristic equation is: $r - 2 = 0$.

Result is therefore: $a_n = C \times 2^n$, where C is a constant.

$a_0 = s_0 - t_0 = 4 - 1 = 3$ using the initial conditions. Consequently, $a_0 = C \times 2^0 = C$, we deduce that $C = 3$, i.e., $a_n = 3 \times 2^n$.

It follows:

$$s_n = a_n + t_n = 3 \times 2^n + t_n,$$

and we can now substitute s_n by this last expression in the first recurrence relation of the system.

We obtain:

$$3 \times 2^n + t_n = 8(3 \times 2^{n-1} + t_{n-1}) - 9t_{n-1}.$$

After some algebraic manipulations, we get:

$$t_n + t_{n-1} = -3 \times 2^n + 3 \times 8 \times 2^{n-1} = 3^2 \times 2^n.$$

Characteristic equation: $r + 1 = 0$, therefore the general expression of the solution for the homogeneous part is:

$$t_n = A \times (-1)^n,$$

where A is a constant.

Particular solution is of the form: $C \times 2^n$. After substitution, we get $C = 6$.

It follows that the general expression of the solution for recurrence relation (2) is:

$$t_n = A \times (-1)^n + 6 \times 2^n.$$

Using the initial condition $s_0 = 4$ leads to:

$$t_0 = 4 = A \times 1 + 6, \qquad \text{then} \qquad A = -2, \qquad \text{hence} \qquad t_n = -2 \times (-1)^n + 3 \times 2^n.$$

We deduce:

$$s_n = a_n + t_n = 3 \times 2^n - 2 \times (-1)^n + 3 \times 2^n = -2 \times (-1)^n + 3 \times 2^{n+1}.$$

Exercise 2.15

The analysis of an algorithms leads you to the following recurrence relation:

$$a_n = a_{n/2} + a_{n/4} + n \qquad n \geq 2, \qquad a_1 = a_2 = a_3 = 1.$$

Give an asymptotic solution to this recurrence.

Solution

Change of variable: $n = 2^k$, and $b_k = a_{2^k} = a_n$. Resulting recurrence relation: $b_k = b_{k-1} + b_{k-2} + 2^k$.

Characteristic equation: $x^2 - x - 1 = 0$, hence 2 roots:

$$x_1 = \frac{1+\sqrt{5}}{2} \qquad \text{and} \qquad x_2 = \frac{1-\sqrt{5}}{2}.$$

General solution of the homogeneous recurrence relation:

$$b_k = C_1 \times \left(\frac{1+\sqrt{5}}{2}\right)^k + C_2 \times \left(\frac{1-\sqrt{5}}{2}\right)^k.$$

Particular solution: is of the form $C_3 \times 2^k$. After substitution, we get $C_3 = 4$.

General solution of the non homogeneous recurrence relation:

$$b_k = C_1 \times (\frac{1+\sqrt{5}}{2})^k + C_2 \times (\frac{1-\sqrt{5}}{2})^k + 4 \times 2^k.$$

Initial conditions: $b_0 = a_{2^0} = a_1 = 1 = C_1 + C_2 + 4$, that is $C_1 = 1 - C_2 - 4 = C_2 - 3$.

$$\begin{aligned} b_1 = a_{2^1} = a_2 = 1 = C_1 \times \frac{1+\sqrt{5}}{2} + C_2 \times \frac{1-\sqrt{5}}{2} + 4 \times 2 \\ = (C_2 - 3) \times \frac{1+\sqrt{5}}{2} + C_2 \times \frac{1-\sqrt{5}}{2} + 4 \times 2. \end{aligned} \tag{2.29}$$

We next need to come back to a_n:

$$a_n = b_{\log_2 n} = ...$$