DM582 Exercises - Sheet 9

Mads Anker Nielsen Tobias Samsøe Sørensen

March 19, 2026

This document contains exercises from the course DM582 (spring 2025). Most exercises are from the book *Introduction to Algorithms, 4th edition* by Cormen, Leiserson, Rivest, and Stein (CLRS), the book *Algorithm Design, 1st edition* by J. Kleinberg and E. Tardos (KT), and the book *Discrete Mathematics and its Applications, 8th edition* by K. Rosen.

References to CLRS refer to the book *Introduction to Algorithms, 4th edition* by Cormen, Leiserson, Rivest, and Stein.

References to KT refer to the book *Algorithm Design, 1st edition* by J. Kleinberg and E. Tardos.

References to Rosen refer to the book *Discrete Mathematics and its Applications, 8th edition* by K. Rosen.

References to BG refer to the book *Computer Algorithms: Introduction to Design and Analysis, 3rd edition* by Sara Baase and Allen Van Gelder.

This document will inevitably contain mistakes. If you find some, please report them to your TA so that we can correct them.

Sheet 9

CLRS, 32.1-1

Exercise

Show the comparisons the naive string matcher makes for the pattern 0001 in the text 000010001010001.

CLRS, 32.1-2

Exercise

Suppose that all characters in the pattern are different. Show how to accelerate `NAIVE-STRING-MATCHER` to run in time $O(n)$ on n-character text T.

CLRS, 32.1-3

Exercise

Suppose that pattern and text are randomly chosen strings of length m and n respectively, from the d-ary alphabet $\Sigma_d = \{0, 1, \ldots, d-1\}$, where $d \geq 2$. Show that the expected number of character-to-character comparisons made by the implicit loop in line 4 of the naive algorithm is

$$(n - m + 1)\frac{1 - d^{-m}}{1 - d^{-1}} \leq 2(n - m + 1)$$

over all executions of this loop. (Assume that the naive algorithm stops comparing characters for a given shift once it finds a mismatch or matches the entire pattern.) Thus, for randomly chosen strings, the naive algorithm is quite efficient.

CLRS, 32.1-4

Exercise

Suppose we allow the pattern P to contain occurrences of a gap character $\diamond$ that can match an arbitrary string of characters (even one of zero length). For example,

Example omitted. See CRLS page 962.

The gap character may occur an arbitrary number of times in the pattern but not at all in the text. Give a polynomial-time algorithm to determine whether such a pattern P occurs in a given text T, and analyze the running time of your algorithm.

CLRS, 32.2-1

Exercise

Working modulo 11, how many spurious hits does the Rabin-Karp matcher encounter in the text 3141592653589793 when looking for the pattern $P = 26$?

CLRS, 32.2-4

Exercise

Alice has a copy of a long n-bit file $A = (a_{n-1}, A_{n-2}, ..., A_0)$, and Bob similarly has an n-bit file $B = (b_{n-1}, b_{n-2}, ..., b_0)$. Alice and Bob wish to know if their files are identical. To avoid transmitting all of A or B, they use the following fast probabilistic check. Together, they select a prime $q > 1000n$ and randomly select an integer x from $\{0, 1, ..., q-1\}$. Letting

$$A(x) = \sum_{i=0}^{n-1} a_i x^i \mod q \quad \text{and} \quad B(x) = \sum_{i=0}^{n-1} b_i x^i \mod q,$$

Alice evaluates $A(x)$ and Bob evaluates $B(x)$. Prove that if $A \neq B$, there is at most one chance in 1000 that $A(x) = B(x)$, whereas if the two files are the same, $A(x)$ is necessarily the same as $B(x)$. (Hint: See Exercise 31.4-4.)

CLRS, 32.3-1

Exercise

Draw a state-transition diagram for the string-matching automaton for the pattern $P = aabab$ over the alphabet $\Sigma = \{a, b\}$ and illustrate its operation on the text string $aaababaabaababaab$.