Algorithm Engineering
2 February 2022 – time 60 minutes

Question #1 [scores 5+5]. Given the string T = abababc,

- compress by LZ77
- compress by LZ78, showing the auxiliary data structure used.

Question #2 [scores 6]. Given the string S = "amata$" show the result of the compression via the algorithmic pipeline BWT + MTF + RLE0 + Huffman. Assume that MTF counts symbol positions from 0, and RLE0 uses the Wheeler's code.

Question #3 [scores 4+3]. Given the two sored lists L1 = (1, 2, 4, 6, 9, 10, 15, 18, 20) and L2 = (2, 3, 7, 8, 18) compute their intersection using the

- Mutual partitioning strategy
- Two-level storage approach, with block size b=3 for the list L1

Question #4 [scores 4]. Assume you are given a set of 4 strings {aa, ac, bc, cc} and you wish to construct a minimal ordered perfect hash function (MOPHF), where the order is the alphabetic one. Assume that rank(x) =2; 3; 4 for the characters x = a; b; c; respectively. Given a string xy of two characters, we let the two random functions required by the design of MOPHF be h1(xy) = (3 * rank(x) + rank(y)) mod 7 and h2(xy) = (rank(x) + rank(y)) mod 7.

Question #5 [scores 3] Given a text T[1,n], design an algorithm that exploits a Suffix Array built on T to efficiently establish whether: Given a string P[1,p] and two positive integers k and q, there exists a range of k contiguous positions in T, say T[i, i+k-1], where **start** q occurrence of P in T. In this case it prints "i" otherwise, if no such range does exist, it prints "-1".