This exercise sheet covers different topics of the course to prepare you for the exam. Throughout this exercise sheet, let m be your Matrikelnr. having 8 digits $m_1m_2m_3m_4m_5m_6m_7m_8$ with $0 \leq m_i \leq 9$. **Start each document handed in with (writing down) your name and m.**

1) a) For $k = m_7 + 2$, consider k-mergesort, the variation on mergesort that sorts a list ℓ of length greater than 1 by splitting it into k lists $\ell_1, \dots, \ell_k$ of equal lengths, recursively k-mergesorts $\ell_1, \dots, \ell_k$ to yield $s_1, \dots, s_k$, and then does a k-way merge of the $s_1, \dots, s_k$ to yield a sorted list s.

   Analyse the (time) complexity $T(n)$ of k-mergesort, for n the length of the input-list. You may restrict your analysis to lists whose length is a power of k (so that in each recursive call all k parts indeed do have equal lengths), and you may assume that a k-way merge takes time linear in the length of the merged lists.

   b) Does there exist a natural number x such that $x \equiv m_2 \pmod{m_3 + 2}$ and $x \equiv m_4 \pmod{m_3 + 3}$? If so, compute such an x by applying Bézout's Lemma/the Chinese Remainder Theorem. If not, argue why not.

   c) Show that the set $LP = \{M\#x \mid \text{TM } M \text{ loops on input } x\}$ is not recursive, where you may assume that (the code of) M and x are bit-strings.

2) a) Let k be the number m_3m_4 and k' be m_5m_6 in decimal notation.

   Compute $\text{lcm}(k, k')$ by first computing $\gcd(k, k')$ by means of the Euclidean algorithm, giving the intermediate steps. (You may choose either the subtraction-based or the division-based version; indicate which version you use.)

   b) Let k be the number $2 + m_2$ in decimal notation.

   Consider all functions from the set $N = \{n \mid 0 \leq n < k \cdot (k+1)\}$ of natural numbers to the set $P = \{(x, y) \mid 0 \leq x < k,\ 0 \leq y < k + 1\}$ of pairs of natural numbers.

   i) Show that $|N| = |P|$ by giving some bijection between N and P.

   ii) How many injective functions from N to P are there? Since the number is typically large, it suffices to explain how it is computed.

   iii) Is the function $f(n) = (n \mod k,\ n \mod (k+1))$ a bijection from N to P? If so, explain why. If not, show how bijectivity fails.

   c) Let k be your matrikelnr. m interpreted as a number in decimal notation. Does there then exist an inverse of k modulo 15? If so, compute the inverse. If not, explain why not. (You may use a calculator for intermediate steps, say for modulo computations, but you have to explain your method.)

3) a) Let G be a directed graph with nodes $\{a, b, c, d\}$ and labeled edges

$$\{(a, m_2, b), (a, m_3, c), (a, m_4, d), (b, m_5, a), (b, m_6, a), (c, m_7, d), (d, m_8, c)\}$$

   where each triple describes the start of the edge, the weight, and the end of the edge. Use Floyd's algorithm to compute the distances (least weight among possible paths) between all nodes. Give the start matrix and all intermediate matrices, and give the distance from b to d.

b) Let G be the graph with nodes $\{a, b, c, d, e, f, g\}$ and edges

$$\{a, b\}, \{a, e\}, \{b, e\}, \{c, d\}, \{c, g\}, \{d, g\}, \{d, f\}, \{f, g\}$$

The weights of the edges in this order are $m_1, m_2, m_3, m_4, m_5, m_6, m_7, m_8$. Use Kruskal's algorithm to compute a spanning forest with minimal weight. Show all the intermediate steps and the final spanning forest.

c) Consider the relation R on digits $\{0, \ldots, 9\}$, defined by $R(d, e)$ if and only if no instance of the digit d appears anywhere after any instance of the digit e in the word m. Formally:

$$R = \{(d, e) \mid 0 \leq d, e \leq 9,\ \nexists u, v, w.\ m = uevdw\}$$

For example, in the word 121, we have $2R2$, $3R4$, $\neg 1R1$, $\neg 1R2$, $\neg 2R1$.

Is R an equivalence relation? If so, prove it. If not, give for **each** property (among the 3 properties equivalence relations have) that is not satisfied a counterexample.

4*) a) Prove the following statement if it is true or give a counterexample if it is false: Let M, N be countably infinite sets such that $M \subseteq N$. Then $N \setminus M$ is finite.

b) For languages L_1, L_2 over Σ, L_1 is said to be *reducible* to L_2, denoted by $L_1 \leq L_2$, if there exists a computable $f : \Sigma^* \to \Sigma^*$ such that $x \in L_1 \Leftrightarrow f(x) \in L_2$.

Explain how the notion of reducibility is typically used to show languages are not recursive, and why the condition that f be *computable* cannot be omitted from the definition (of reducibility), without rendering it useless for that usage.

c) Show that the language $L = \{M\#x \mid \text{TM } M \text{ never moves to the left on input } x\}$ is recursive.