1. Presence Exercise Sheet for the Lecture
Computer Science Theory

Exercise 1: Sets
(a) Two sets are equal if and only if _______________________________.

Write down all elements (without the duplicates) for the twelve sets below.

1: ________  2: ________  3: ________
4: ________  5: ________  6: ________
7: ________  8: ________  9: ________
10: ________ 11: ________ 12: ________

Draw lines between those sets which are equal.

1  2  3
∅  {♦, ♥}  {}

4  {♦♥}  5  {{♣}}

6  {{}, ∅}  7  {{♣, ♣}}

8  {{}}  9  {♥, ♦}

{∅}  {♣}  {∅, ♦}  
10  11  12
(b) Apply the following set operations and give the number (i), yes/no (ii–iii), and the resulting sets (iv–vi).

(i) $|S|$ for finite set $S$ is defined as ________________.

|{}| = ____
|{♥, ♠}| = ____
|{}| = ____
|{♥, ♠}| = ____

(ii) $e \in S$ if and only if ________________.

{} ∈ {}
{} ∈ {♥, {}}
♦ ∈ {♥, {♦}}
{♥} /∈ {♥, ♠}

(iii) $S_1 \subseteq S_2$ if and only if ________________.

{} \subseteq {}
{} \subseteq {{♦}}
{♦, ♠} \subseteq {♦, ♥, {♣}}
{♦, ♠} \subseteq {♠, ♥, ♦}

(iv) $S_1 \cup S_2$ is the set which ________________.

{} \cup {} = ______
{} \cup {♥} = ______
{♦, ♥} \cup {♥, ♠} = ______

(v) $S_1 \cap S_2$ is the set which ________________.

{} \cap {} = ______
{} \cap {♥} = ______
{♦, ♥} \cap {♥, ♠} = ______

(vi) $S_1 \setminus S_2$ is the set which ________________.

{♠} \setminus {} = ______
{} \setminus {♥} = ______
{♦, ♥} \setminus {♥, ♠} = ______
Exercise 2: Natural numbers as a language
Consider the alphabet $\Sigma = \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$.

Give a definition for a language $L$ over $\Sigma$ containing exactly all natural numbers ($\mathbb{N}$) without leading zeros. This means we do not want to have 1 and 001 (but only 1).

**Hint:** You can define the language directly or you can apply set operations.
Ask yourself: are we interested in how many leading zeros a word has?
Do not forget 0 (zero).

Exercise 3: Deterministic finite automata
Consider the following picture:

We start at 1 and want to get to 3. We can move from 1 to 2, from 2 to both 1 and 3, and from 3 to 2.

(a) Your task is to represent the language of all valid moves from 1 to 3 as a DFA.

   (i) Let $\Sigma = \{r\}$. For each $r$ we go one step to the right.
   
   (ii) Let $\Sigma = \{\ell\}$. For each $\ell$ we go one step to the left.
   
   (iii) Let $\Sigma = \{r, \ell\}$.

(b) How many words are accepted in each case?

(c) How can we modify the automaton from (iii) if

   (i) we start at 2?
   
   (ii) we want to get to 2 instead?
   
   (iii) we want to get to 2 or 3?
Exercise 1: Reverse Operator
Consider $\Sigma = \{a, b, c\}$.

(a) What is the language $L(A)$ and its reverse language $L(A)^R$ for the NFA $A$ below?

```
q_0 \rightarrow a \quad q_1 \quad b \quad q_2 \quad c \quad q_3
```

$L(A) = \underline{\text{___________}}$  \hspace{2cm} $L(A)^R = \underline{\text{___________}}$

Construct an NFA that recognizes the reverse language $L(A)^R$.

(b) What is the problem with the construction if we have more than one final state?

Exercise 2: Regular Expressions
Construct regular expressions for the following languages over the alphabet $\Sigma = \{a, b\}$.

(a) $L_1 = \{a, b, ab\}$

(b) $L_2 = \Sigma^*$

(c) $L_3 = \Sigma^+$

(d) $L_4 = \{w \in \Sigma^* | w \text{ starts with } a\}$

Exercise 3: Pumping Lemma
The proof always works as follows:

(a) Assume the language $L$ is regular. Then the pumping lemma must hold.

(b) Assume some $n \in \mathbb{N}$ from the pumping lemma. You must not make any assumptions on $n$.

(c) Smartly choose a word $z \in L$ (usually depending on $n$) with $|z| \geq n$.

(d) Assume some decomposition $z = uvw$ (with the rules given in the pumping lemma).

(e) Smartly choose some $i \in \mathbb{N}$ such that $uvw^i \notin L$ (often $i = 0$ or $i = 2$ suffices).

With this “algorithm” in mind, read the example provided in the lecture script on page 27. In general, you may need to make a case distinction in step (d). Then for each case you have to find some $i$ in the next step. But in this exercise it is not necessary.