Tutorial 2

Exercise 1: Construct and describe in detail a ***universal Turing machine*** $\mathcal{U}$, which expects as an input a word of the form $u\#v$ where u is a code of a one-tape Turing machine $\mathcal{M}$ and v is an input for this machine $\mathcal{M}$.

For simplicity, you can consider just those cases where machines $\mathcal{M}$ is assumed to use the tape alphabet $\Gamma = \{0, 1, \square\}$ and input alphabet $\Sigma = \{0, 1\}$, and v is a word over this alphabet Σ.

What should be modified to allow the machine $\mathcal{U}$ to simulate computations of Turing machines $\mathcal{M}$ with arbitrary tape alphabet Γ and arbitrary input alphabet Σ?

Exercise 2: Explain, what will do a universal Turing machine $\mathcal{U}$ if it obtains a word of the form $u\#v$, where u is a code representing machine $\mathcal{U}$, as an input.

Exercise 3: Consider the following problem:

> NÁZEV: UHP (*Uniform Halting Problem*)
>
> INPUT: Turing machine $\mathcal{M}$.
>
> QUESTION: Does $\mathcal{M}$ halt for every input?

Find out whether this problem is decidable or undecidable, and prove your claim. If the problem is decidable, show an algorithm solving this problem; if it is undecidable, you can use undecidability to show the corresponding reduction.

Exercise 4: Let us recall a ***tiling*** problem descibed in the lecture where it was shown that the problem is undecidable.

This problem can be defined as follows. Let us say that C is a finite set of *colors*. The set $\{N, S, E, W\}$ represents four *directions* — north, south, east, west. A ***type of a tile*** is specified as an assignment of colors to directions, i.e., as a function $\tau : \{N, S, E, W\} \to C$.

Let us assume that we have a set of types of tiles

$$\mathcal{T} = \{\tau_1, \tau_2, \dots, \tau_n\}.$$

Covering of a plane with tiles is a function $p : \mathbb{Z} \times \mathbb{Z} \to \mathcal{T}$ satisfying the following two conditions for each $i, j \in \mathbb{Z}$:

- If $p(i, j) = \tau$ a $p(i + 1, j) = \tau'$, then $\tau(E) = \tau'(W)$.
- If $p(i, j) = \tau$ a $p(i, j + 1) = \tau'$, then $\tau(N) = \tau'(S)$.

Consider now the following problem:

> INPUT: A set of types of tiles $\mathcal{T} = \{\tau_1, \tau_2, \dots, \tau_n\}$.
>
> QUESTION: Is the covering of a plane p with tiles from the set $\mathcal{T}$?

It is known that this problem is undecidable.

Does this problem remain undecidable if there would be an addition restriction that the number of colors, i.e., the size of set C can not be greater than some given constant k (e.g., it is not possible to use more than 100 colors)?

Justify your answer.

Exercise 5: Show that the following problem is undecidable:

INPUT: A pair of Turing machines $\mathcal{M}_1$ and $\mathcal{M}_2$.

QUESTION: Is $\mathcal{L}(\mathcal{M}_1) = \mathcal{L}(\mathcal{M}_2)$?

Is this problem or its complement semidecidable?

Exercise 6: Show that the following problem is undecidable:

INPUT: Context-free grammars $\mathcal{G}_1$ and $\mathcal{G}_2$.

QUESTION: Is $\mathcal{L}(\mathcal{G}_1) \cap \mathcal{L}(\mathcal{G}_2) = \emptyset$?

Hint: You can use a reduction from the Post correspondence problem.

Is this problem or its complement semidecidable?

Exercise 7: Show that the following problem is undecidable:

INPUT: A context-free grammar $\mathcal{G}$.

QUESTION: Is $\mathcal{G}$ ambiguous?

Hint: You can use a reduction from the Post correspondence problem.

Is this problem or its complement semidecidable?

Exercise 8: Consider the following two problems:

INPUT: A context-free grammar generating a language over an alphabet Σ.

QUESTION: Is $\mathcal{L}(\mathcal{G}) = \Sigma^*$?

INPUT: Context-free grammars $\mathcal{G}_1$ and $\mathcal{G}_2$.

QUESTION: Is $\mathcal{L}(\mathcal{G}_1) = \mathcal{L}(\mathcal{G}_2)$?

a) Show a reduction from the first of these problems to the second.

b) Show how the Halting problem can be reduced to the complement of the first problem.

c) Determine, which of these problems or their complements are semidecidable.

Exercise 9: Recall how formulas of the first-order predicate logic are defined, and what it means that a given formula is closed.

Recall also what it means that a given formula is ***logically valid***, i.e., true in every interpretation.

Show that the following problem is undecidable:

> INPUT: A closed formula φ of the first-order predicate logic.
>
> QUESTION: Is formula φ logically valid?

Hint: You can use a reduction from the Post correspondence problem.

Exercise 10: A ***linear bounded automaton*** is a special case of a one-tape Turing machine where its tape is not infinite but it is restricted to the size of the input word. The tape consists of cells that contain an input word w where special endmarks $\vdash$ and $\dashv$are added to the left and to the right. On the left endmark $\vdash$, the head can not move to the left, and on the right endmark $\dashv$, the head can not move to the right. These endmarks can not be overwritten but all other cells (that contain at the beginning the symbols of the input word) can be.

The language $\mathcal{L}(\mathcal{M})$ of words accepted by the given linear bounded automaton $\mathcal{M}$ is defined similarly as in the case of standard Turing machines.

Consider the following two problems:

> INPUT: Linear bounded automaton $\mathcal{M}$ and word w.
>
> QUESTION: Does the automaton $\mathcal{M}$ accepts the word w, i.e., is $w \in \mathcal{L}(\mathcal{M})$?

> INPUT: Linear bounded automaton $\mathcal{M}$.
>
> QUESTION: Is there some word w accepted by the automaton $\mathcal{M}$, i.e., is $\mathcal{L}(\mathcal{M}) \neq \emptyset$?

Determine, which of these problems are decidable and which are not.

For the problems that are undecidable, determine where the given problem or its complement are semidecidable.

Exercise 11: Consider a machine with a control unit with finite number of states and with one counter. This machine reads its input from an input tape. This input tape is read-only, so its content can not be changed. The head on this input tape can be moved in both directions and the word is bounded from the left and from the right by endmarkes $\vdash$ and $\dashv$.

The counter can contain an arbitrary natural number as a value. In one step, the value of the counter can be increased by 1, decreased by 1, or unchanged, and it is also possible to test if the value of the counter is 0.

Define formally this kind of a machine, and define what it means that this machine accepts a given word w.
Show that the following problem is undecidable:

INPUT: One-counter nachine $\mathcal{M}$.

QUESTION: Is there a word w accepted by machine $\mathcal{M}$, i.e., is $\mathcal{L}(\mathcal{M}) \neq \emptyset$?

Exercise 12: Give an example of an undecidable problem that is reducible to complement.

Exercise 13: Give examples of at least three properties of Turing machines whose undecidability follows from Rice's theorem, and at least three properties whose undecidability does not follow from Rice's theorem.