Finite automata are mathematical objects being one of the simplest computational models. Such an automaton, at any moment, stores only its current state from finitely many possible ones. Receiving various input signals, it can change the current state to another one. The number of possible input signals is of course also finitely many. Therefore, a finite automaton can be described as a list of states and rules precising for every state and input signal to what state the automaton should move.

In real world, finite automata can be found in many different places. For example, the behavior of a typical washer is based on a finite automaton: in a simplified way, its states are "off", "on", "pre-wash", "wash", "rinse", and "spin". The input signals for a washer are pressing buttons and also a signal informing about elapsing a time amount, which causes to switch to the next stage. Of course, modern washers have a much greater set of states connected with possible choices of various options: they have such states as "pre-wash, selected the cotton program", "pre-wash, selected the synthetics program" etc. In fact, it can be said that every computer is a finite automaton, because finally their number of possible states (the content of memory) is finitely many (although it is a very large number), as well as possible actions from the user.

Finite automata find applications as models in such fields as engineering, electronics, linguistics, biology, and philosophy. They are very often used in programming (e.g. parsers, schemes of behavior of an artificial intelligence). In computer science they are a subject of extensive theoretical studies. Also, they have strong connections with mathematics, in particular with semigroup theory.

In computer science, in the theory of formal languages, it is assumed that an automaton answers to some question: it reads an input data and says "yes" or "no". The input data sequences that are accepted by an automaton we call *language*.

Within this project we plan to study selected classical problems in the theory of finite automata and their applications. The underlying conception is to use new ideas and methods that appeared during the recent studies.

The studies will concern the so called *avoiding words*, which could be used to make a progress in a longstanding open mathematical problem – the Černý conjecture from 1969. The conjecture states that if we do not know the current state of the automaton, then we can bring the automaton to one state that we know by applying a sequence of at most $(n-1)^2$ input signals, if such a sequence exists at all (where n is the number of states of the automaton).

We will also study universality problems – how to detect whether a given automaton always says "yes" for every input data. This is one of the most basic and well studied questions that we can ask having an automaton. Despite theoretical importance, it has applications in such fields as knowledge representation and databases. But what if we are interested only in some lengths of input data? Or if it is enough that the automaton accepts not necessarily all, but sufficiently many different input data? Within the project we would like to solve such problems and extend available knowledge.

Moreover, we will work on an application of the languages of finite automata in the field of Artificial Intelligence, in particular to describe rules of an arbitrary board game. It turns out that automata (actually, their languages) are very good in this role, and they can describe formally even complicated rules in such games like chess and checkers. A general language describing board games would be used in such areas as General Game Playing – creating a program that can play any given board game and understand its rules, and also in computer generating of rules of new games and automatic learning of rules by observing. From the practical point of view, such a language must be both rich enough in expressiveness and also be friendly for computing. Therefore, it is necessary to build theoretical foundations, which will be a goal of the project and which will allow us to find a well balanced solution from the perspective of applications.