1 Research project objectives/Research hypothesis The main topic of the project is the grammar compression, in which we represent the data, treated as a string of letters (string), as a context free grammar generating this string. The increasing popularity of grammar compression is a result of a couple of reasons: First, it is practical: the grammar compressed data is of similar size the one compressed by other popular compression formats. Second, due to its natural inductive definition it is easy to further process it: we can perform varied operations on the grammar-compressed data without the need of explicit decompression. Third, the algorithm processing compressed data turned out to be useful in other areas of computer science: for many problems (with the most known example being the satisfiability of word equations) the best solution is to compute (or nondeterministically guess) the compressed representation and perform the operations directly on it.

Still, there are many open questions regarding the grammar compression, in this project we are going to investigate some of them:

Grammar compressors and entropy. Empirical entropy is a standard measure of how complex a string is, but its connection to grammar compression remains unknown; the applicant wants to establish this connection and employ it in a construction of a grammar compressor and in the evaluation of known heuristical compressors.

Approximation ratio of grammar compressors. The general problem of construction of the smallest grammar for a string is NP-hard and it is not known how well it can be approximated. We want to improve lower bounds on this problem and establish the approximation ratios of known heuristical compressors.

Running time lower bounds. The most often used algorithm for grammar compressed strings is the equivalence check for two grammars, i.e. whether they define the same word. The currently best implementation has quadratic running time. The applicant wants to establish whether there is a matching or almost matching lower bound; we also plan to give other running time lower bounds for algorithms on grammar compressed strings.

Forests grammars. There are more sophisticated grammar compression models, for instance for trees. Their main advantage is that they preserve the tree structure during the compression and they allow larger class of operations on the compressed representation. However, there are a couple of models of such a compression, we want to investigate a newly presented one, the so called *forest grammars*: compare it with the pre-existing models, verify possible applications and extend some known techniques and results to them.

Applications of grammar compression. The applicant plans to give new algorithms using grammar compression for some open problems connected to word equations and term rewriting.

2 Research project methodology The varied field of study and research topics imply that a large number of different methodologies are used; below the most prominent ones are listed.

Recompression. One of the most recent methodology for compressed data. It is based on analysing and modification of the compressed representation, while ignoring the uncompressed structure. Applied successfully in construction of upper-bounds and algorithms.

Word Combinatorics. The oldest and classic methodology for string algorithms, in which we analyse and exploit the combinatorial structure of the strings: periodicity, repetitions, pattern avoidance *etc*. Recently there were first successful attempts of combining it with the recompression approach.

Kolmogorov Complexity. The best lower bounds for approximation ratios of concrete algorithms are based on a consequence of Kolmogorov complexity: there exist n-bit strings such that each their description (in particular: by a grammar) is of size at least n (bits).

Conditional lower bounds: SETH. Condition lower bounds are the most recent and strongest ways to show running time lower bounds for various problems. In this methodology we assume some computational complexity hypothesis (usually: Strong Exponential Time Hypothesis) and with this assumption we prove that the running time of any algorithm for a given problem has to be at least some given bound, as otherwise SETH would fail. Such bounds were recently shown for a couple of problems related to compressed data, we plan to give such bounds to more such problems, in particular to the most important one: the equivalence of the grammar compressed representations.

3 Expected impact of the research project on the development of science Grammar compression is an important topic in computer science, both theoretical and applied one. The proposed topics are central and difficult in this area and due to applications of grammar compression they have implications for other areas of computer science. Better algorithms or compression models yield immediate applications and automatically lead to improvements in known applications. Better models potentially allow applications to problem which were not covered by previous models. On the other hand, lower bounds exhibit that potential improvements need can only be found elsewhere. The analysis of practical heuristics provides solid arguments backing their performance. Yet, as this is a theoretical work, the most important is the understanding of this model itself.