Results;

Regex engine, which is developed by the Group 21, named as G21\_Regex, is compared to the regex engines which can be developed by different groups and company. In this study, two important parameters, duration and matching status, are analyzed and compared with G21\_Regex.

Regex engines, analyzed in this project, are the most used and well-known on the internet. They are available with different support of expressions, performance constraints and language bindings. Comparison of the regex engines was assumed to be started in 2003 with John Maddock, who is part of Boost developers. For different regular expressions the time taken to find all occurrences of the expression within a long English language text was measured. This text is consisted of 750.000 lines and approximately 20million characters. After that comparison, sljit project repeated this benchmark with same input file and different regex expressions in 2015. In 2017 Rust Leipzig enlarged the benchmark with different regex engine libraries.

In this study, the aim is to give an overview of actively developed engines regarding their performance and comparison with each other and our regex engine. TRE is a lightweight, robust, and efficient POSIX compliant regexp matching library with some exciting features such as approximate (fuzzy) matching. Boost Regex in one of the first regex engine which has capability to cope with wide character strings, or search and replace operations, something that traditional C libraries can not do. RE2 uses finite-state machine using automata theory, in contrast to almost all other regular expression libraries, which use backtracking implementations. Perl Compatible Regular Expressions (PCRE) is inspired by the capabilities of the Perl programming language. Engine developed by Philip Hazel in 1997. PCRE's syntax is much more powerful and flexible than either of the POSIX regular expression flavors and than that of many other regular-expression libraries. Oniguruma is a regular expressions library which encompasses features from different regular expression implementations that traditionally exist in different languages and supports a variety of character encodings.

Table

Description automatically generated

These libraries are developed by professional software developers. Aim of this study is to develop a compatible regex engine in matching status, but not in terms of duration. Results for developed regex engines and our engines are tabulated in the Table.X. It can be easily seen that G21\_Regex engines has capability of finding regex in the string. On the other hand, It is needed to optimize in terms of time.