**National University of Computer and Emerging Sciences, Karachi  
FAST School of Computing, Spring 2025**

**Theory of Automata CS301**

**Project Proposal:** Text Search Engine

**1. Project Overview**

* **Project Title:** Text Search Engine
* **Course:** Theory Of Automata
* **Instructor:** Miss Shahar bano

**2. Group Members**

* **Group Member 1:** [Huzaifa Altaf, 23K-0627]
* **Group Member 2:** [Waleed Hussain, 23K-0885]
* **Group Member 3:** [Shayan Nemat, 23K-0899]

**3. Project Description:**

The project involves the development of a **regular expression-based text search engine**. The system will allow users to enter a regular expression and search for words that match a specific pattern in a given text file. Additionally, users can replace matched words using the replace functionalities.

**Objectives:**

1. Develop a search engine capable of finding words in a text file based on a given regular expression.
2. Implement error handling for incorrect or malformed regular expressions.
3. Enable replacement of matched words with user-specified replacements.
4. Design an interactive UI that allows users to input regular expressions and perform search and replace operations.

**Scope of the Project:**

* **Concepts Covered:**
  + Finite Automata (DFA/NFA)
  + Regular Expressions
  + String Matching Algorithms
  + File Handling
* **Expected Outcomes:**
  + Users can enter a regular expression and search for words that match the given pattern.
  + Users can find all occurrences of a matching word in the file.
  + Users can replace one or all occurrences of a matching word with a given replacement.
  + The system provides an interactive UI for input and output display.
* **Functional Features:**
  + **Find:** Displays the first word in the file that matches the input regular expression.
  + **Find All:** Displays all words in the file that match the input regular expression.
  + **Replace:** Replaces the first matched word with the specified replacement word.
  + **Replace All:** Replaces all matched words with the specified replacement word.
  + **View File:** Displays the current content of the file.

**4. Methodology:**

The project will be developed from scratch using programming concepts from automata theory. The implementation will include parsing regular expressions and constructing finite automata to match words efficiently.

**Tools and Technologies Used:**

* **Programming Language:** C++
* **Regex Library:** regex (C++)
* **GUI Framework:** Qt (C++)
* **File Handling:** Standard file I/O operations

**5. Team Contributions:**

**Group Member 1:**

* Handles the functionality for inputting regular expressions.
* Implements error handling for invalid expressions.

**Group Member 2:**

* Implements **Find, Find All, Replace, and Replace All** functionalities.
* Develops file handling mechanisms for searching and replacing words.

**Group Member 3:**

* Designs the UI including buttons, fonts, and layout.
* Implements UI rendering and integrates background images.

**6. Challenges and Risks:**

* **Handling Complex Regular Expressions:** Advanced regex features may be difficult to implement efficiently.
* **Performance Optimization:** Large text files may slow down searches; optimizations may be needed.
* **UI Responsiveness:** Ensuring the UI remains interactive while performing file operations.
* **Error Handling:** Avoiding crashes due to incorrect or complex regex patterns

**7. References:**

* Introduction to Computer Theory by Cohen
* Miss Shaharbano’s Slides
* Various online resources on DFA and NFA implementations

**8. Conclusion:**

This project will provide a hands-on implementation of automata theory by integrating **regular expressions, finite automata, and file handling** into a practical application. The system will offer a useful tool for text processing and demonstrate the power of **automata theory in pattern matching**. Future improvements may include support for **custom regex operators**, **advanced pattern matching techniques**, and **multi-file search capabilities**.