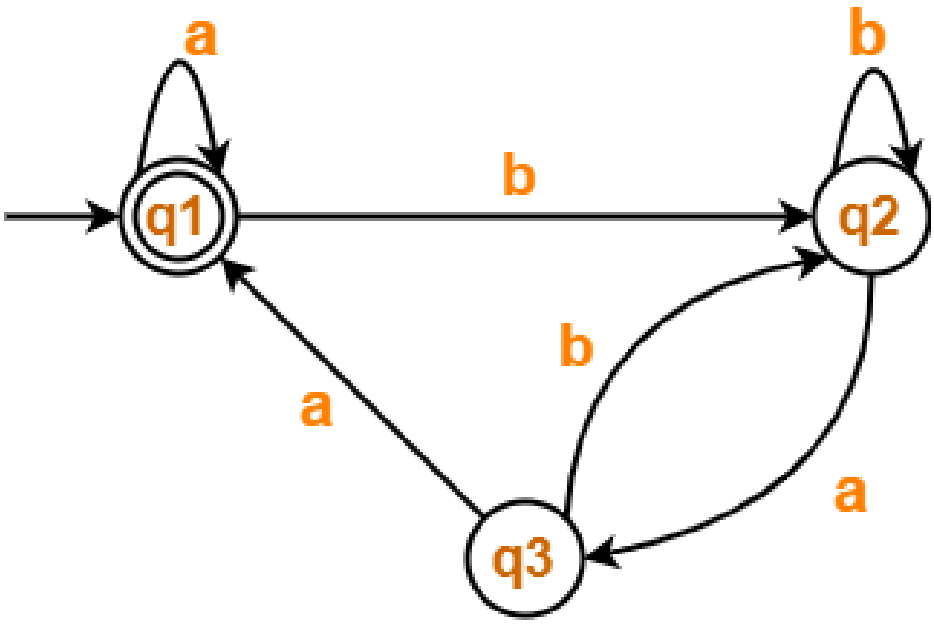
REGEX TO DFA CONVERTER– REPORT



Team Members

Omar Magdy – 231000967

Nourhan Elsheikh – 231000853

Heba Youssef – 231000723

Reem Ali – 231000165

Project Description

This project provides a fully interactive tool that converts a Regular Expression (Regex) into a deterministic finite automaton (DFA).
The system helps users understand the behavior of automata step-by-step by
allowing them to convert, and test input strings directly on the generated
DFA.
The input to the system is a valid regular expression, and upon conversion,
the user can enter a string to check whether it is accepted or rejected by the
resulting DFA.

Input Format

User enters a valid regular expression such as: aabb

Output Format

• Acceptance or rejection of a test input string

Inside Mechanism

The system performs the full theoretical process of constructing finite automata:
1. Converts a given Regex into postfix form using operator precedence.
2. Builds an NFA using Thompson's Construction algorithm.
3. Subset Construction Applies to convert the NFA to DFA.
4. Simulates input strings through the DFA and determines acceptance.

Programming Languages, Tools & Libraries

- Python
- Collections & Math Libraries

Screenshots & Output Samples

Screenshots & Output Samples

Input: aabb

```
PS E:\NU\SEMESTER1\THEORY\Project-phase1> & C:/Users/hyous/AppData/Local/Microsoft/WindowsApps/python3.13.exe e:/NU/SEMESTER1/THEOR
Y/Project-phase1/Regex-to-DFA-Converter/finalversion.py
 Welcome to the Regex → DFA Converter & Simulator
---------------------------------------------------
Please enter your regular expression: aabb

 Processing your regex...
Regex accepted successfully!
Postfix form: aa.b.b.

DFA Transition Table:
------------------------------------------------------------
State     |     a     |     b
------------------------------------------------------------
S0        |     S1    |     -     |
S1        |     S2    |     -     |
S2        |     -     |     S3    |
S3        |     -     |     S4    |
S4        |     -     |     -     |
------------------------------------------------------------
Start State: S0
Accepting State(s): S4

 Your DFA is ready! Now you can test any string below:
---------------------------------------------------
```

```
  Your DFA is ready! Now you can test any string below:
  ---------------------------------------------------

   Enter a string to test (or type 'exit' to quit): aa
  Rejected!

   Enter a string to test (or type 'exit' to quit): abab
  Rejected!

   Enter a string to test (or type 'exit' to quit): aabb
  Accepted!

   Enter a string to test (or type 'exit' to quit): bb
  Rejected!

   Enter a string to test (or type 'exit' to quit): baba
  Rejected!
```

Input: (a|b)*abb

```
PS E:\NU\SEMESTER1\THEORY\Project-phase1> & C:/Users/hyous/AppData/Local/Microsoft/WindowsApps/python3.13.exe e:/NU/SEMESTER1/THEORY/Project-phase1/Reg
DFA-Converter/finalversion.py
 Welcome to the Regex → DFA Converter & Simulator
----------------------------------------------------------
Please enter your regular expression: (a|b)*abb

 Processing your regex...
Regex accepted successfully!
Postfix form: ab|*a.b.b.

DFA Transition Table:
------------------------------------------------------------------
State      |     b      |     a
------------------------------------------------------------------
S0         |     S1     |     S2     |
S1         |     S1     |     S2     |
S2         |     S3     |     S2     |
S3         |     S4     |     S2     |
S4         |     S1     |     S2     |
------------------------------------------------------------------
Start State: S0
Accepting State(s): S4
```

```
Your DFA is ready! Now you can test any string below:
----------------------------------------------------

 Enter a string to test (or type 'exit' to quit): aabb
Accepted!

 Enter a string to test (or type 'exit' to quit): abab
Rejected!

 Enter a string to test (or type 'exit' to quit): ababb
Accepted!

 Enter a string to test (or type 'exit' to quit): bbab
Rejected!

 Enter a string to test (or type 'exit' to quit): bbabb
Accepted!

 Enter a string to test (or type 'exit' to quit): baabb
Accepted!

 Enter a string to test (or type 'exit' to quit): exit
```

Input: (a|b)*a(a|b)

```
er/finalversion.py
 Welcome to the Regex → DFA Converter & Simulator
----------------------------------------------------------
Please enter your regular expression: (a|b)*a(a|b)

 Processing your regex...
Regex accepted successfully!
Postfix form: ab|*a.ab|.

DFA Transition Table:
------------------------------------------------------------------
State      |     b      |     a
------------------------------------------------------------------
S0         |     S1     |     S2     |
S1         |     S1     |     S2     |
S2         |     S3     |     S4     |
S3         |     S1     |     S2     |
S4         |     S3     |     S4     |
------------------------------------------------------------------
Start State: S0
Accepting State(s): S4, S3

 Your DFA is ready! Now you can test any string below:
----------------------------------------------------

 Enter a string to test (or type 'exit' to quit): aabb
Rejected!

 Enter a string to test (or type 'exit' to quit): aaab
Accepted!

 Enter a string to test (or type 'exit' to quit): abab
Accepted!
```