

regex to nfa



Rwan Ashraf 221001757

omar elhossiny 221001028

sherif ahmed 221001968

Merna samir 221002000

Zaynab essam 221001623

🔎 **What does it do?**

The Regex-to-NFA Converter is a tool that processes a regular expression and converts it into its equivalent Non-Deterministic Finite Automaton (NFA) using Thompson’s Construction. Additionally, the project supports converting the NFA into a Deterministic Finite Automaton (DFA) using the Subset Construction Algorithm. The tool provides visual representations of both NFA and DFA using Graphviz, HTML, CSS, JavaScript website.

📥 **Input Format:**

* A valid regular expression using characters: a-z, A-Z, 0-9, |, \*, +, ?, ., (, )
* Example: a (b | c)\*d

**📤 Output Format:**

* PNG images of the generated NFA and DFA state diagrams at local machine.
* Example filenames:
  + 📄 nfa\_a\_b\_c\_d\_.png
  + 📄 dfa\_a\_b\_c\_d\_.png
* Processing steps showing input regex prepared infix and postfix notation
* NFA information : includes each transition with it’s next state
* Visualization of NFA and DFA

**⚙️ Inside Mechanism:**

**✅** Regex Parsing

* Converts infix expressions to postfix notation using the Shunting Yard Algorithm.
* Validates input characters and operator precedence.

✅ NFA Construction

* Implements Thompson’s Construction to build the NFA.
* Supports operators:
  + \* (Kleene Star) **⭐**
  + + (One-or-more) ➕
  + ? (Optional) ❓
  + | (Union) 🔀
  + . (Concatenation) **🔗**

**✅** DFA Conversion

* Uses the Subset Construction Algorithm 📊
* Eliminates epsilon (ε) transitions and ensures determinism.

**✅** Graph Visualization

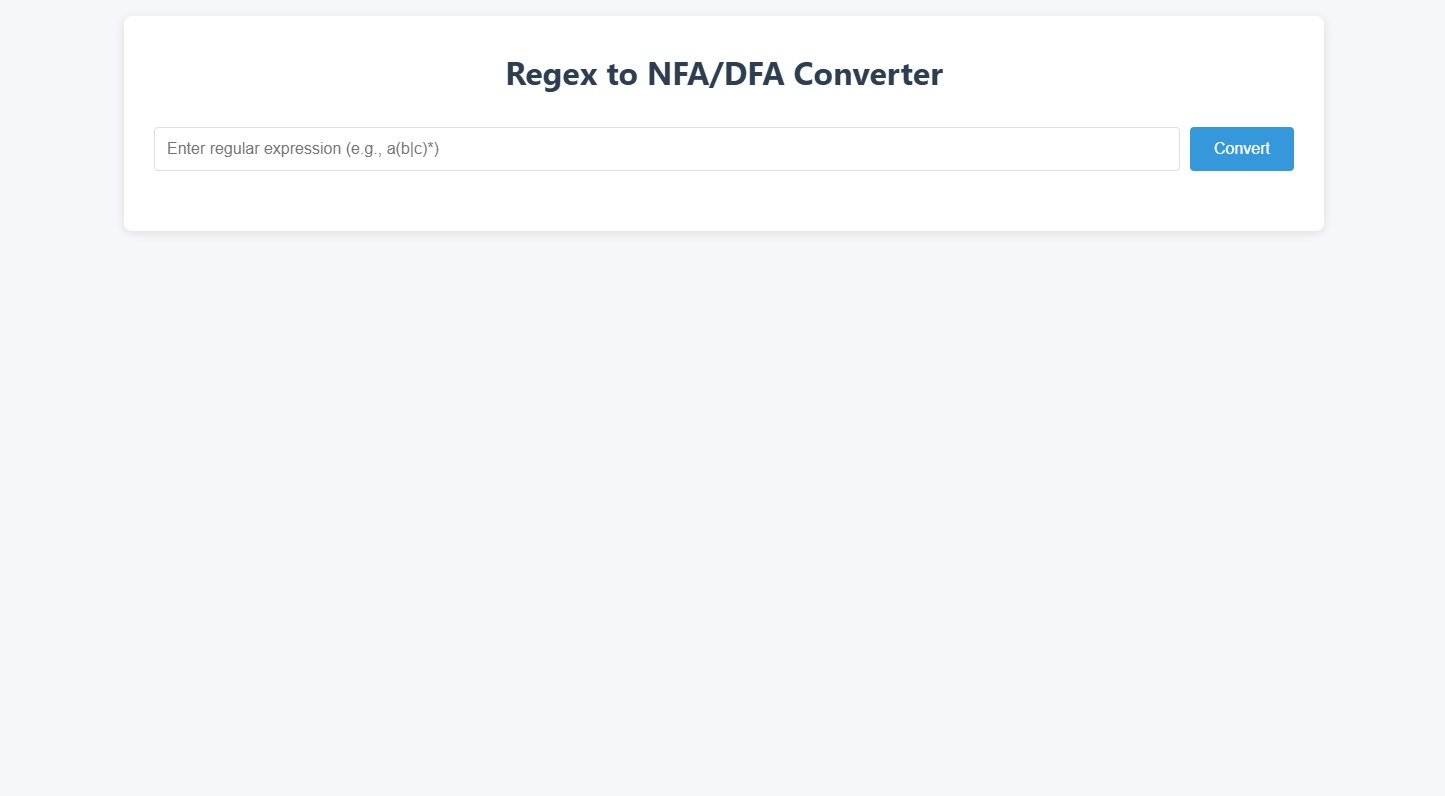
* Uses Graphviz to render and save visual representations of the NFA and DFA 🎨
* Using HTML, CSS and JavaScript for visualizing NFA, transition table and DFA

**💻 Programming Language, Tools & Libraries Used**

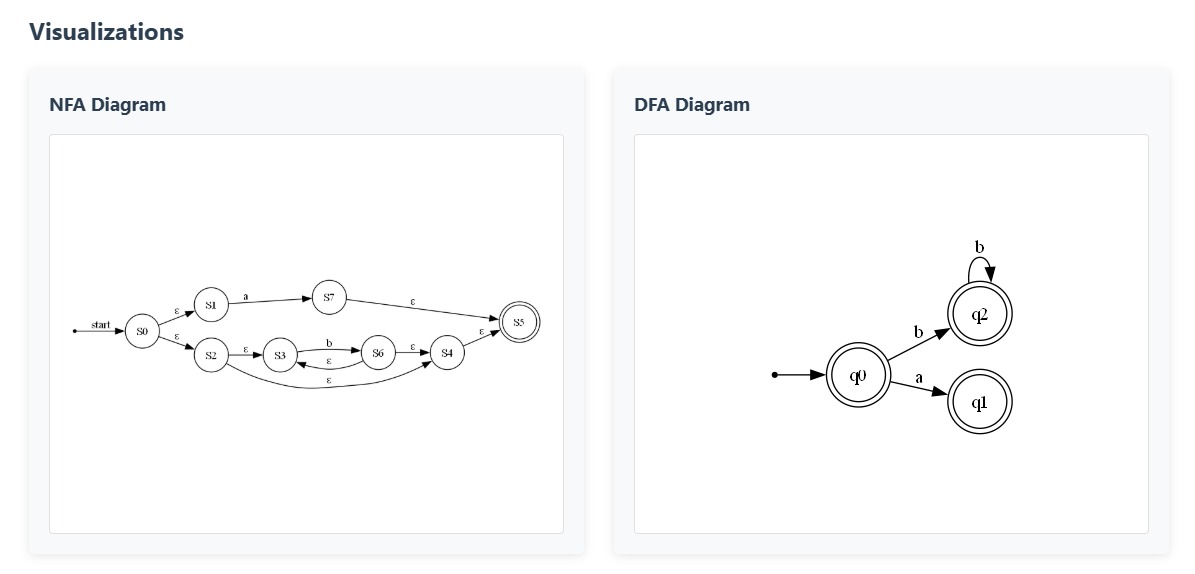
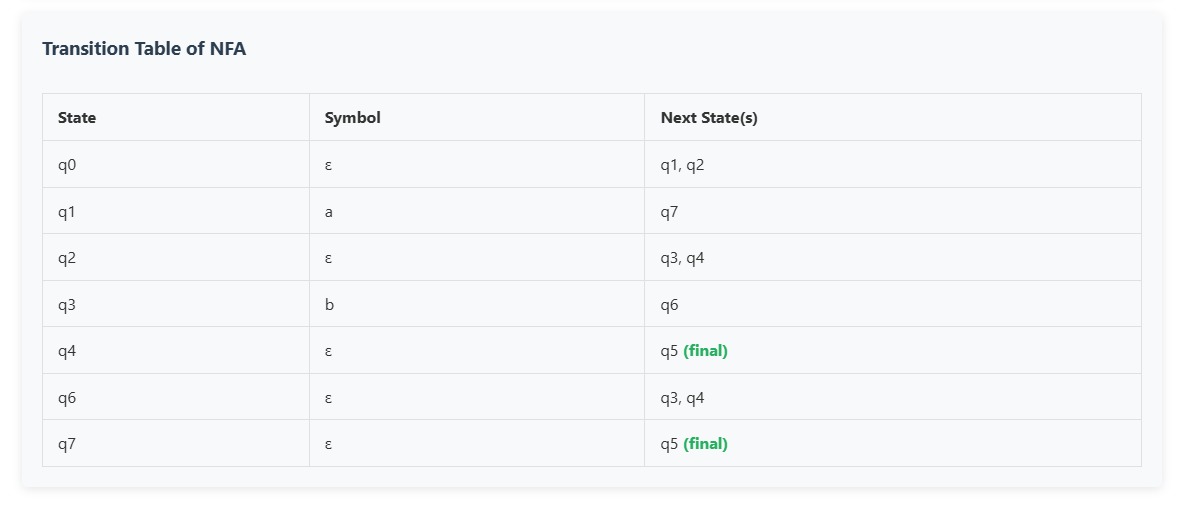
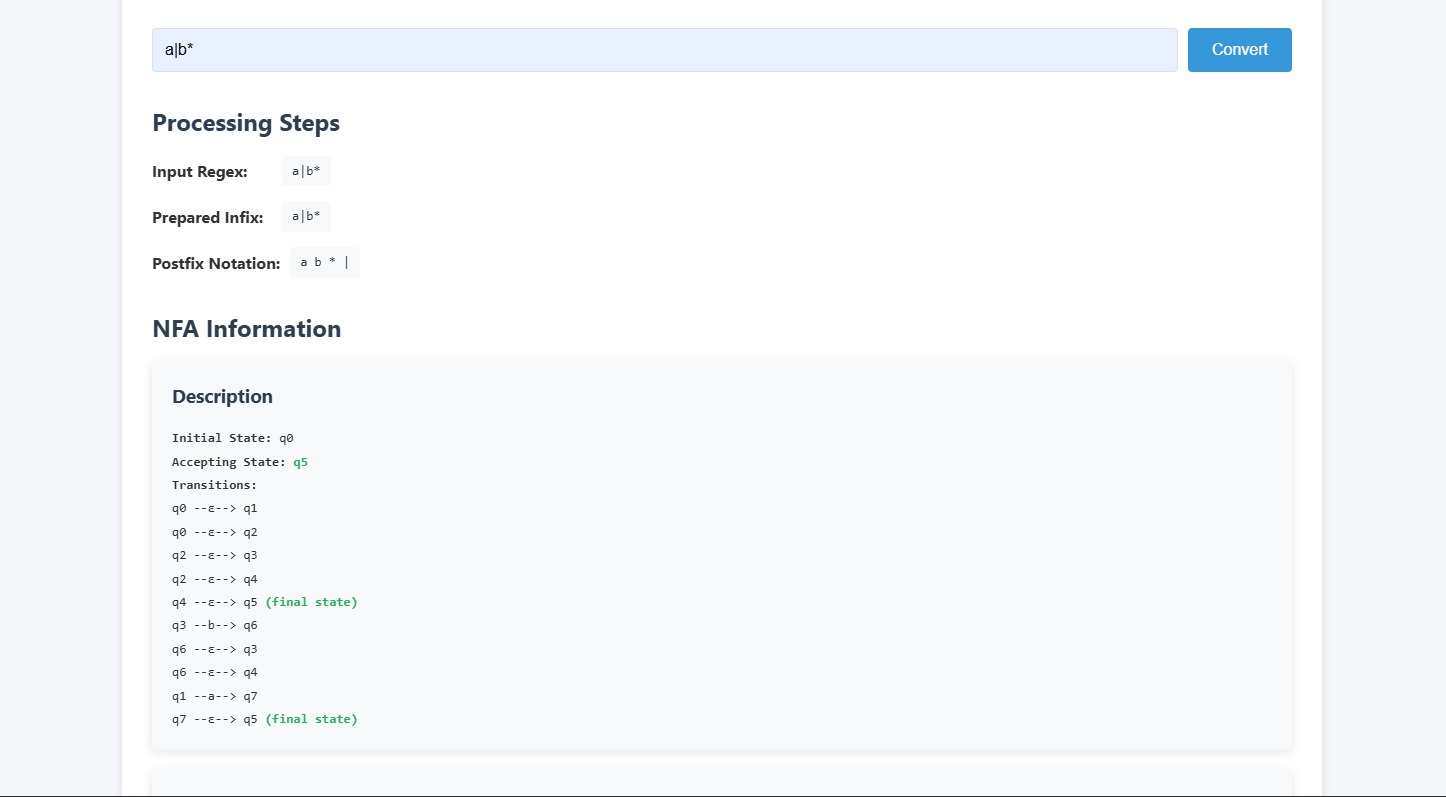
| 🛠 Component | 🚀 Tool/Library Used |
| --- | --- |
| Regex Parsing | Custom Python Parser 🐍 |
| NFA Generation | Thompson’s Algorithm 📐 |
| DFA Conversion | Subset Construction 🔍 |
| Graph Rendering | graphviz Python library 📊 |
| CLI Interface | Python input() function ⌨️ |
| Web UI | HTML, JavaScript, CSS 🌐 |

**📸 Images of the project with the output:**

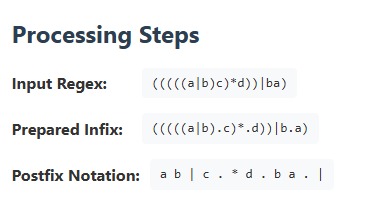
**📥 Input Format:**

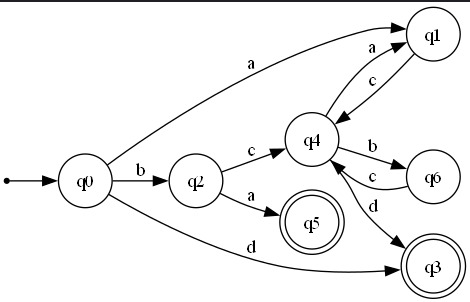
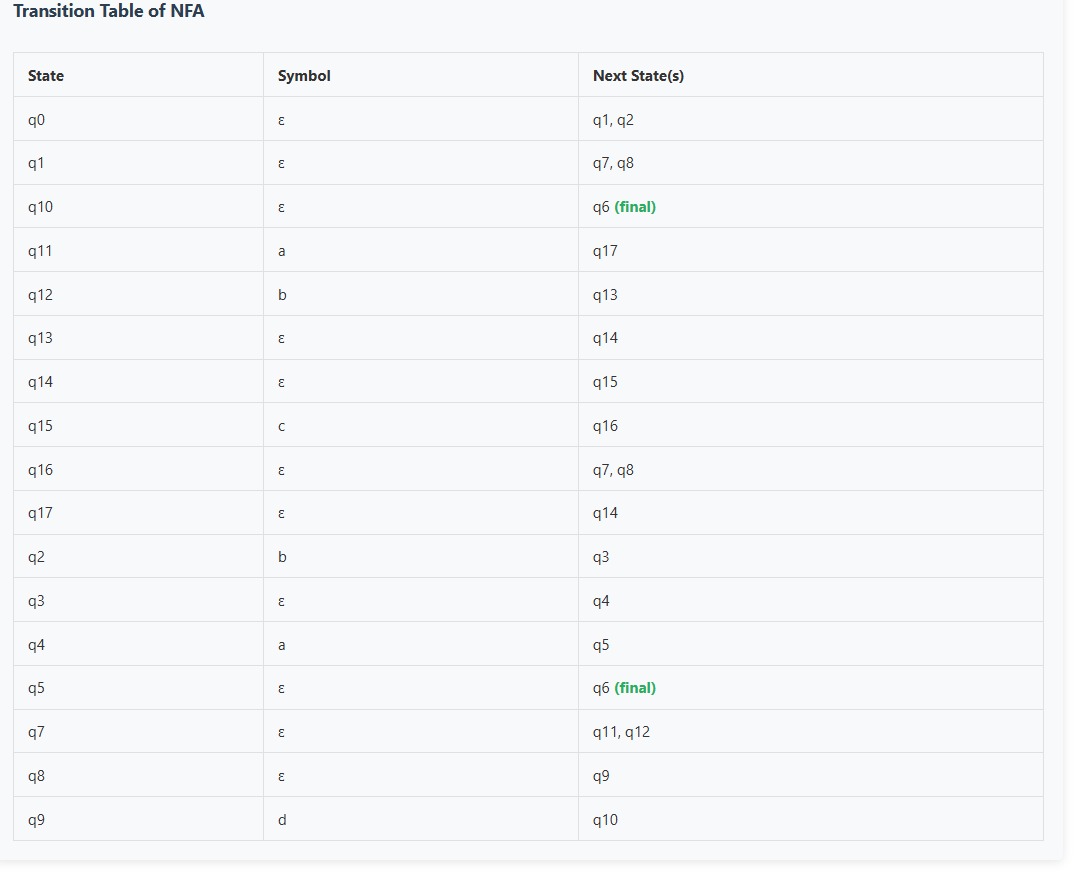


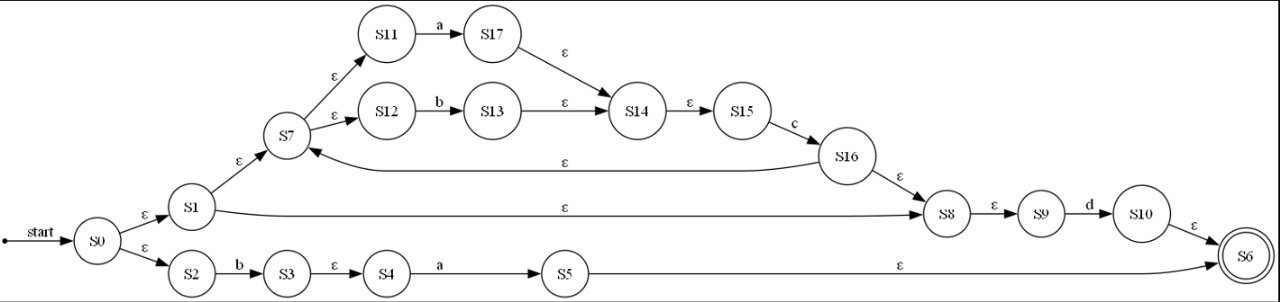
**✅Parsing complex regex:**

**✅ Example 1 with website:**

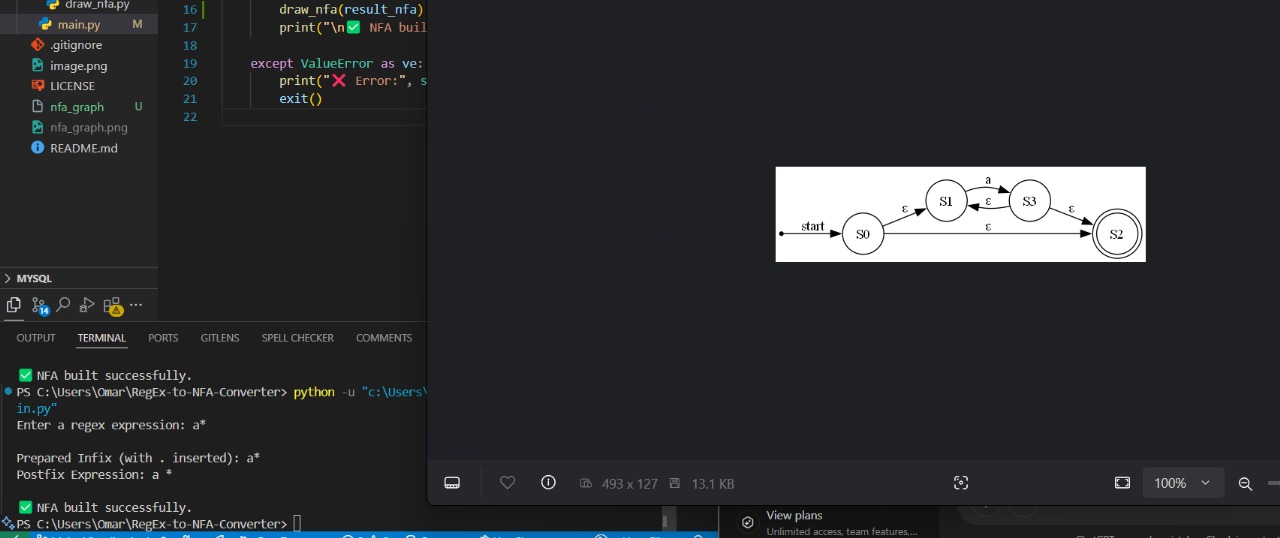
**✅Example 2 with website:**







**✅Example a\* with .png**

****

**🔗 Project Resources & Documentation**

📌 GitHub Repository: [**https://github.com/omar-28-2/RegEx-to-NFA-Converter**](https://github.com/omar-28-2/RegEx-to-NFA-Converter)

📌 Notion Page: [**https://Notion/Regex-to-NFA-Converter**](https://furtive-dietician-ff7.notion.site/REGEX-TO-NFA-Converter-1c81adc35e658074ba47c8100a938a9e)