Alexis Nguyen & Katherine Vu

Regex to NFA Converter

Step 1: Enter Regex

Enter in a regular expression on the command line when running the code. The regular expression must be enclosed in quotation marks.

Example input: python AST_to_NFA “a|b”

Step 2: Read Regex

Parse through the expression to determine if it is in the correct format. If it has mismatched parentheses, double * or | in a row, or invalid characters, then print out an error message.

Example “a|”: Invalid regular expression. Expression in incorrect format

Step 3: Make AST

Recursively iterate through the regex and make an abstract syntax tree with all variables as leaves and the operations as the nodes.

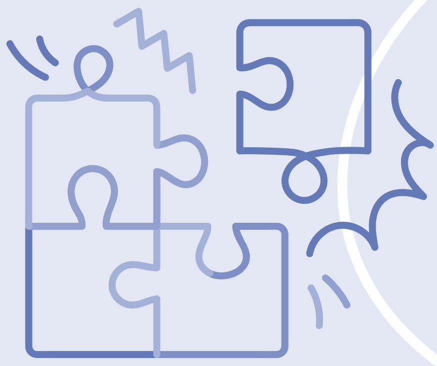
```
Example “a|b”:  ⟶  AST:
                      Or:
                          Leaf: a
                          Leaf: b
```

Step 4: Make NFA

For each node and leaves in the abstract syntax tree, make an NFA that corresponds with each operation and create a start state, accept state, states, alphabet and transition tuple.

```
a: {[q0, q1], [a], {q0: {a: [q1]}}, q0, [q1]}
b: {[q2, q3], [b], {q2: {a: [q3]}}, q2, [q3]}
```

Step 5: AST to NFA



Combine all the nodes into one NFA with a final formal description and print out to a JSON file

```
States: [q4, q0, q1, q2, q3, q5],
Alphabet: [a, b],
Transitions: {q4: {ε: [q0, q2]}, q0: {a: [q1]}, q1: {ε: [q5]},
                    q2: {b: [q3]}, q3: {ε: [q5]}},
Start State: q4,
Accept States: [q5]
```