

Lab 4 Final

https://github.com/cinnamonbreakfast/flcd/tree/main/lab4_2

Data represented on:

States: List<String>

Alphabet: List<String>

Transitions: Dict<String, List<String>> # some sort of key: pair

Final states: List<String>

(Using functional programming, no classes)

Using bold instead of quotation marks:

letter ::= "A" | "B" | ... | "Z" | "a" | "b" | ... | "z"

digit ::= "0" | "1" | ... | "9"

node ::= letter | letter{letter}{digit}

automata ::= line

read ::= **#states** | **#alpha** | **#initial** | **#final** | **#trans**

trans ::= node, node, node

alpha ::= {letter,} letter

line ::= read | {node,} node | trans | alpha