

Compiler Construction
Assignment-01
Automata Design Document
Imaad Fazal 23I-0656 CS-C

Regular Expressions for Each Category:

Keywords: (start|finish|loop|condition|declare|output|input|function|return|break|continue|else)

Identifiers: [A-Z][a-z0-9]{0,30}

Integer Literals: [+-]?[0-9]+

Floating-Point Literals: [+-]?[0-9]+\.[0-9]{1,6}([eE][+-]?[0-9]+)?

Boolean Literals: (true|false)

String Literals: \"([^\"]\\\\\\\\)*\"

Character Literals: '([^\"]\\\\\\\\)'

Arithmetic Operators: (\+|\-|*|\/|\%|*|*)

Relational Operators: (==|!=|<|=|>|<|>)

Logical Operators: (&&|\\\|!)

Assignment Operators: (=|\+=|\-=|*=|\/=)

Inc/Dec Operators: (\+|+|--)

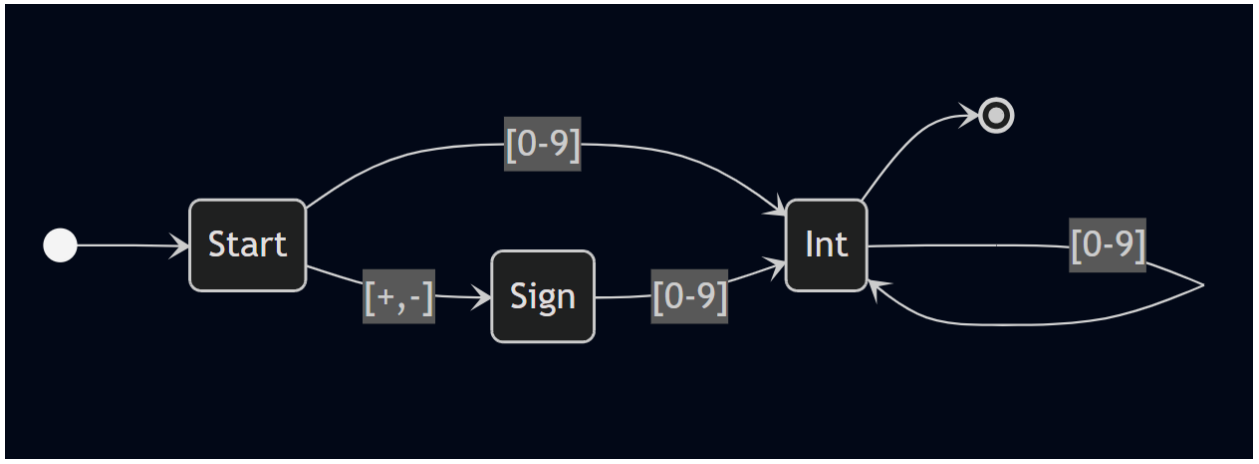
Punctuators: (\(|\)|\{\}|\[|\]|,|;|:)

Single-line Comment: ##[^\\n]*

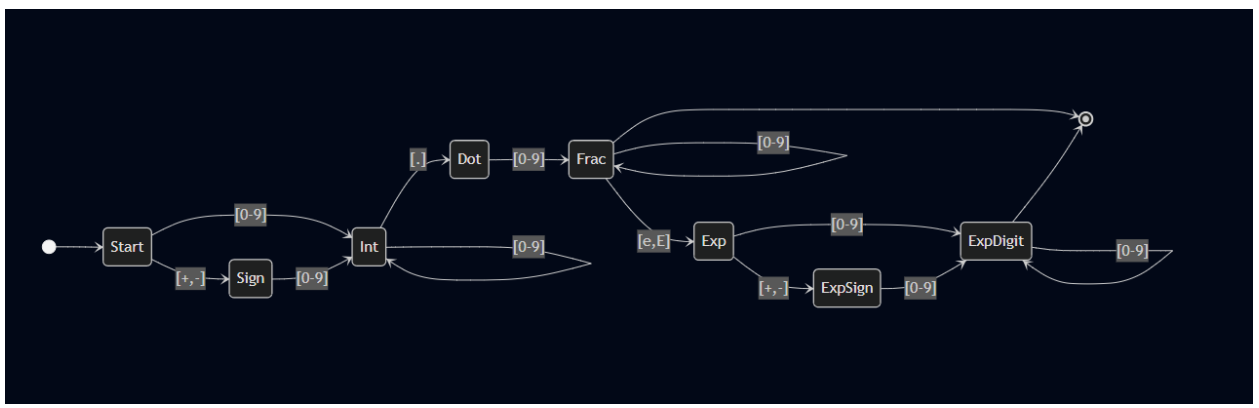
Multi-line Comment: `#*([^*]\|*\^[^\#])**+##`

NFA Diagrams:

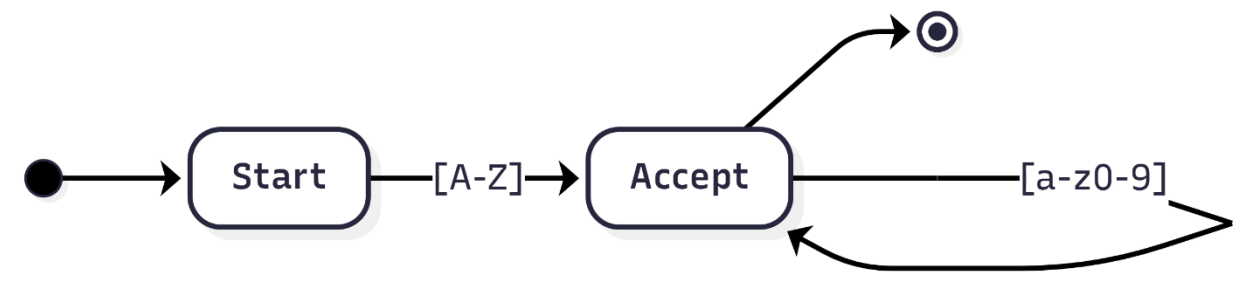
1) Integer Literal (Mandatory):



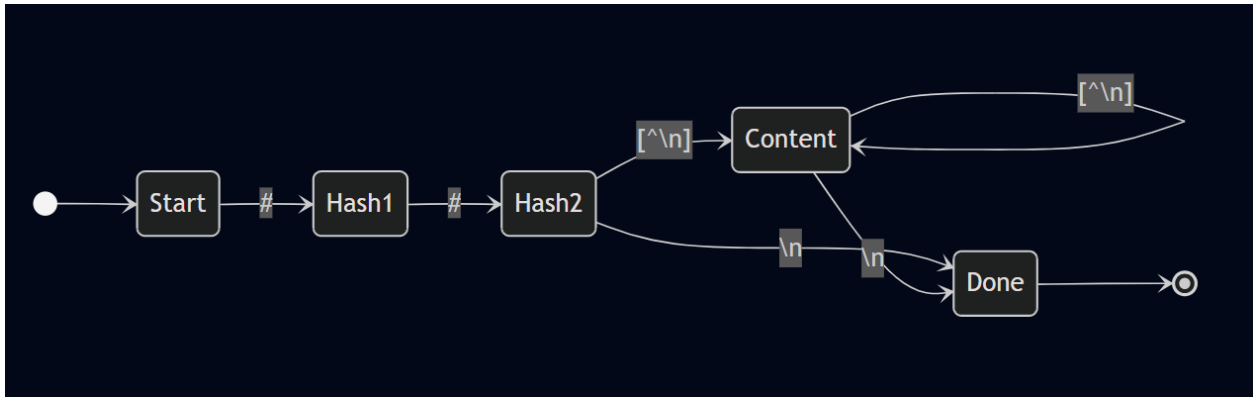
2) Floating Point Literal (Mandatory):



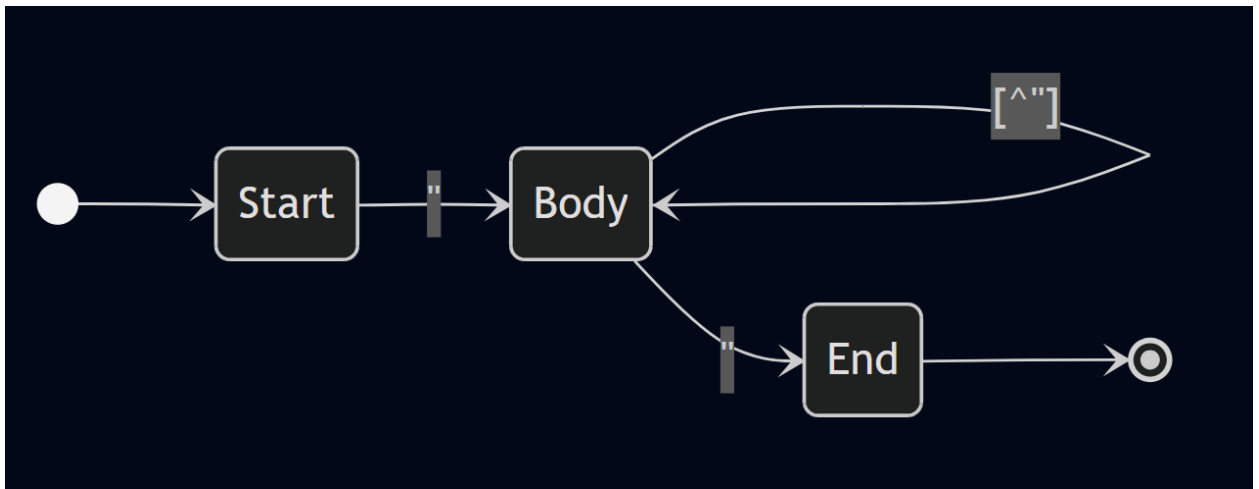
3) Identifier (Mandatory):



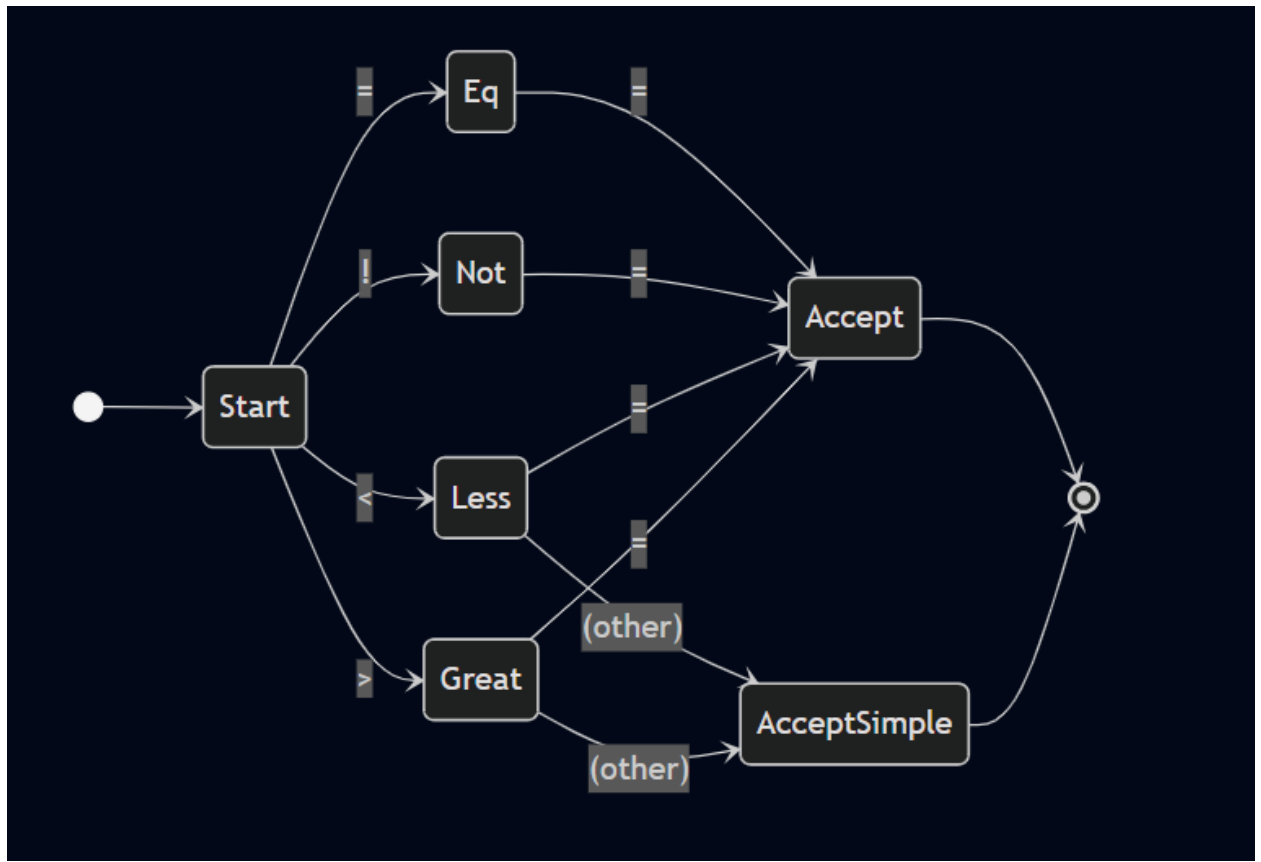
4) Single Line Comment (Mandatory):



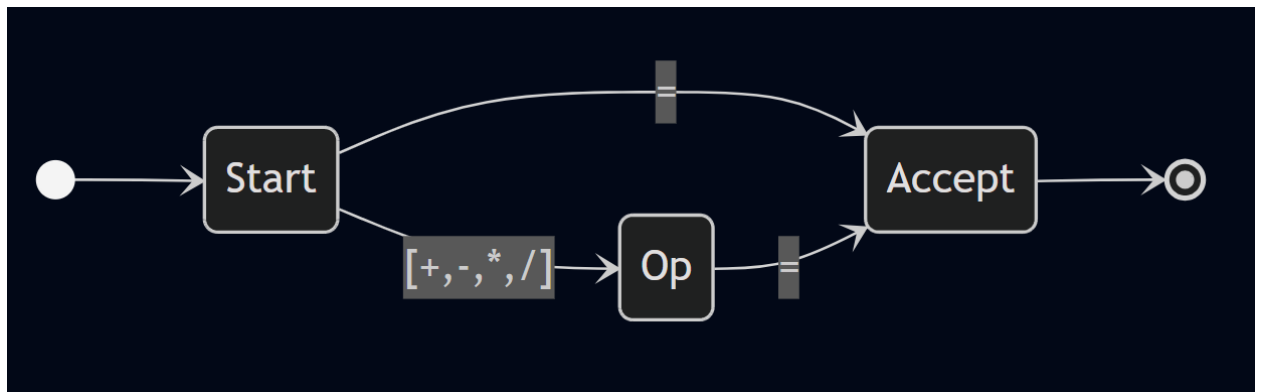
5) String Literal (Choice 1):



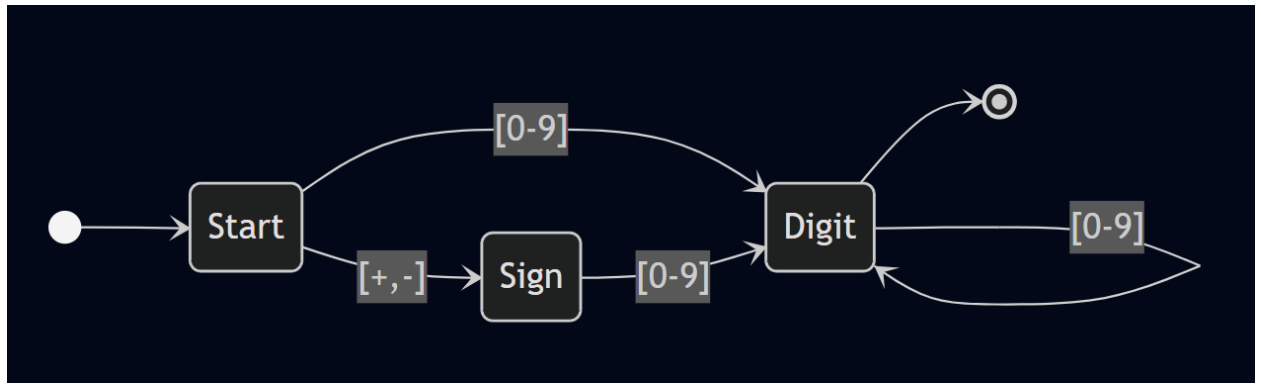
6) Relational Operators (Choice 2):



7) Assignment Operators (Choice 3):



8) Minimized DFA for Integer Literal:



Transition Tables:

1) Identifier (Mandatory):

| State | [A-Z] | [a-z0-9_] | Other |
|--------|--------|-----------|-----------|
| Start | Accept | Error | Error |
| Accept | Accept | Accept | (Retract) |

2) Integer Literal (Mandatory):

| State | [+ , -] | [0-9] | Other |
|-------|---------|-------|-----------|
| Start | Sign | Digit | Error |
| Sign | Error | Digit | Error |
| Digit | Error | Digit | (Retract) |

3) Floating Point Literal (Mandatory):

| State | [+ , -] | [0-9] | . (Dot) | [e , E] | Other |
|-------|---------|---------|---------|---------|-------|
| Start | Sign | IntPart | Error | Error | Error |
| Sign | Error | IntPart | Error | Error | Error |

| State | [+ , -] | [0-9] | . (Dot) | [e , E] | Other |
|----------|---------|----------|---------|---------|-----------|
| IntPart | Error | IntPart | Dot | Error | (Retract) |
| Dot | Error | FracPart | Error | Error | Error |
| FracPart | Error | FracPart | Error | Exp | (Retract) |
| Exp | ExpSign | ExpDigit | Error | Error | Error |
| ExpSign | Error | ExpDigit | Error | Error | Error |
| ExpDigit | Error | ExpDigit | Error | Error | (Retract) |

4) Single Line Comment (Mandatory):

| State | # (Hash) | \n (Newline) | Other Char |
|---------|----------|--------------|------------|
| Start | Hash1 | Error | Error |
| Hash1 | Hash2 | Error | Error |
| Hash2 | Content | Done | Content |
| Content | Content | Done | Content |
| Done | (Accept) | (Accept) | (Accept) |

5) String Literal (Choice 1):

| State | " (Quote) | Other Char |
|-------|-----------|------------|
| Start | Body | Error |
| Body | End | Body |
| End | (Retract) | (Retract) |

6) Relational Operators (Choice 2):

| State | = | ! | < | > | Other |
|-------|-------------|-------|-------|-------|------------|
| Start | Eq | Not | Less | Great | Error |
| Eq | Accept (==) | Error | Error | Error | (Retract) |
| Not | Accept (!=) | Error | Error | Error | Error |
| Less | Accept (<=) | Error | Error | Error | Accept (<) |
| Great | Accept (>=) | Error | Error | Error | Accept (>) |

7) Assignment Operators (Choice 3):

| State | = | +, -, *, / | Other |
|--------|--------------------|------------|-----------|
| Start | Accept (=) | Op | Error |
| Op | Accept (+, -, etc) | Error | (Retract) |
| Accept | (Retract) | (Retract) | (Retract) |