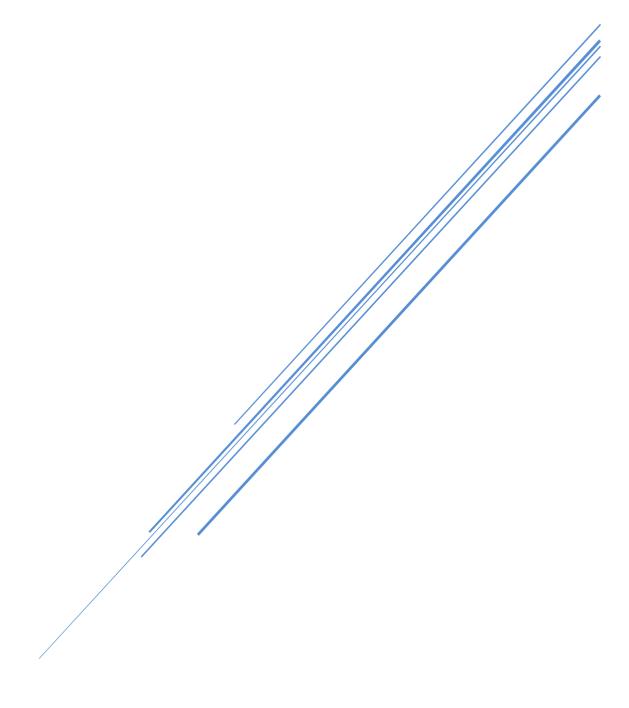
COMPILER THEORY

Tiny Compiler | | MS1

Document containing:

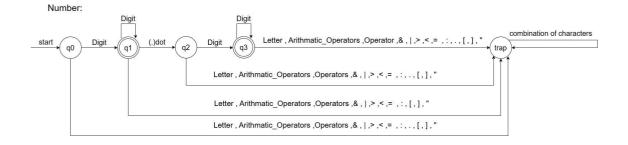
- The Regular Expressions for Tokens.
- DFAs rules of Tokens.



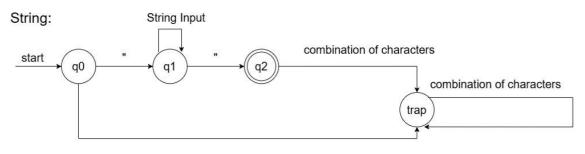
Regular Expressions and DFA for Tiny

Language:

- **Digit:** [0-9]
- Letter: [a-z|A-Z]
- **■ Arithmatic_Operator:** + | | * | \/
- **Operators:** , | ; | \(| \) | { | }
- Combination of characters: Digit | Letter |
 Arithmatic_Operator | Operators |& | \| | > | < | : | = | \. | " |
 \[| \]
- **Number:** Digit+(\.Digit+)?

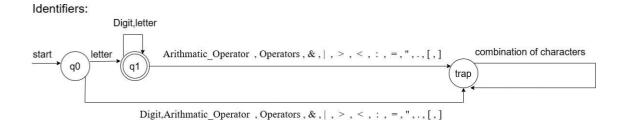


■ **string:** \"(Combination of characters)*\"



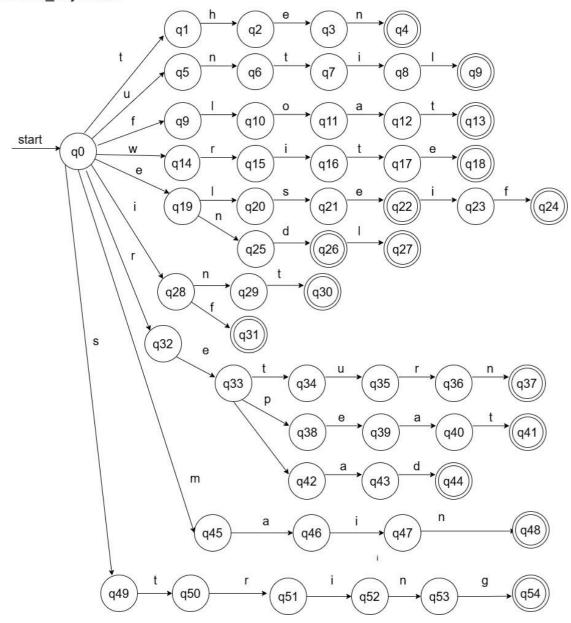
combination of characters

■ Identifiers: Letter (Digit | Letter) *



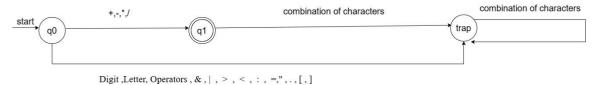
■ **Reserved_Keywords**: int | float | string | read | write | repeat | until | if | elseif | else | then | return | endl | end | main

Reserved_keywords:



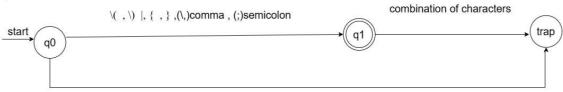
■ Arithmatic_Operator: + | - | * | \/

Arithmatic_Operators:



■ Operators: , | ; | \(| \) | { | }

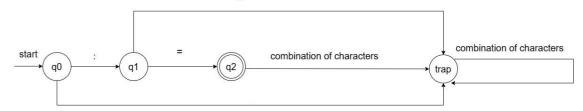
Operators:



Digit ,Letter, Arethmatic_Operator , & , | , > , < , : , =," , . , [,]

Assignment_Operator: :=

$\label{eq:assignment_Operator} Assignment_Operator \ \ \ Digit \ , Letter \ , Arithmatic_Operator \ , Operators \ , \& \ , | \ , > \ , < \ , \ : , ", \dots, [\ , \]$



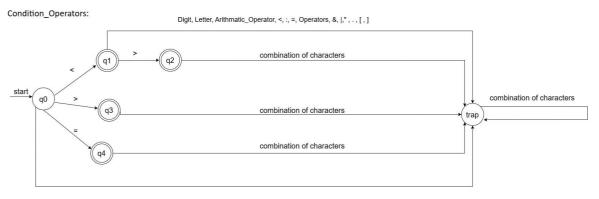
<code>Digit, Letter, Arithmatic_Operator</code> , <code>Operators</code> , & , $|\ ,>\ ,<\ ,=\ ,"\ ,.\ ,[\ ,\,]$

■ Boolean_oprators: && $| \ \ | \$

Boolean_operators Digit, Letter, Arithmatic_Operator, <, >, :, =, Operators, ", . , [,],| start q0 & combination of characters combination of characters combination of characters pligit, Letter, Arithmatic_Operator, <, >, :, =, Operators, ", . , [,], &

 $\label{eq:def:Digit} \mbox{Digit, Letter, Arithmatic_Operator, <, >, :, =, Operators,", . \, , [\ , \]}$

■ Condition_Operator: <|>| <>|=



Digit, Letter, Arithmatic_Operator, :, Operators, &, |," , . , [,]