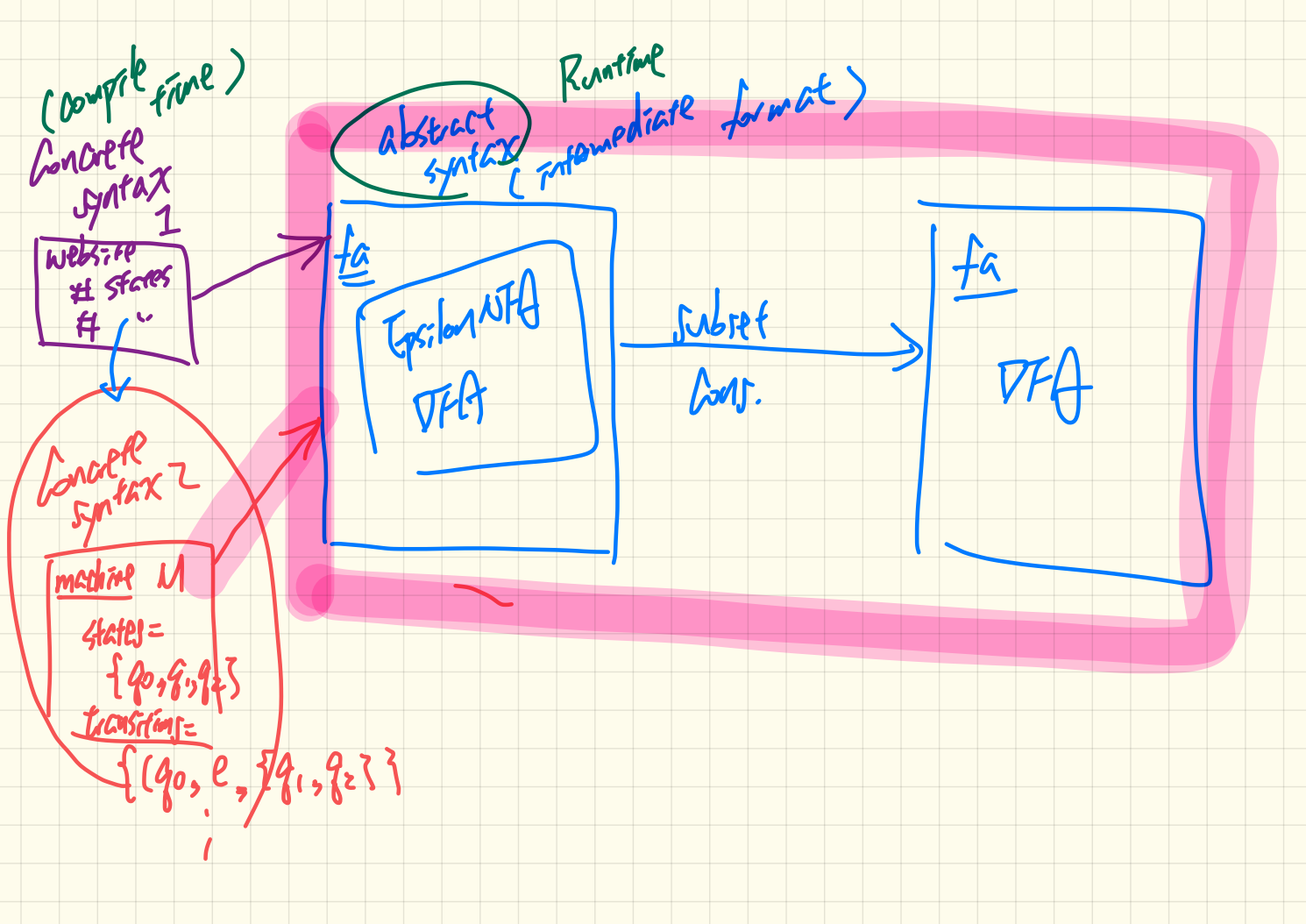
Lecture 7
Monday January 27

transitions

$q_0 : \$ > q_1$

$\{q_0, q_3\} : \$ > q_1$ X

0


(compile time)
Concrete syntax 1
website
states
#
Concrete syntax 2
machine M
states =
{q0, q1, q2}
transitions =
{(q0, e, {q1, q2}}
abstract syntax
(intermediate format)
Runtime
fa
Epsilon NFA
DFA
subset
cons.
fa
DFA

From RE to Scanner (1)

Regular Expression: r[0..9]+

Token Type (CharCat)

r	0, 1, 2, ..., 9	EOF	Other
Register	Digit	Other	Other

Transition

	Register	Digit	Other
s_0	s_1	s_e	s_e
s_1	s_e	s_2	s_e
s_2	s_e	s_2	s_e
s_e	s_e	s_e	s_e

Token Type (Type)

s_0	s_1	s_2	s_e
invalid	invalid	register	invalid

```
NextWord()
  -- Stage 1:  Initialization
  state := s0 ; word := ϵ
  initialize an empty stack S ; s.push(bad)
  -- Stage 2:  Scanning Loop
  while (state ≠ se)
    NextChar(char) ; word := word + char
    if state ∈ F then reset stack S end
    s.push(state)
    cat := CharCat[char]
    state := δ[state, cat]
  -- Stage 3:  Rollback Loop
  while (state ∉ F ∧ state ≠ bad)
    state := s.pop()
    truncate word
  -- Stage 4:  Interpret and Report
  if state ∈ F then return Type[state]
  else return invalid
  end
```

Example input: r241

state : s0 s1 s2 s2 s2 se

word : r241

word: r24 | syn. Not: register

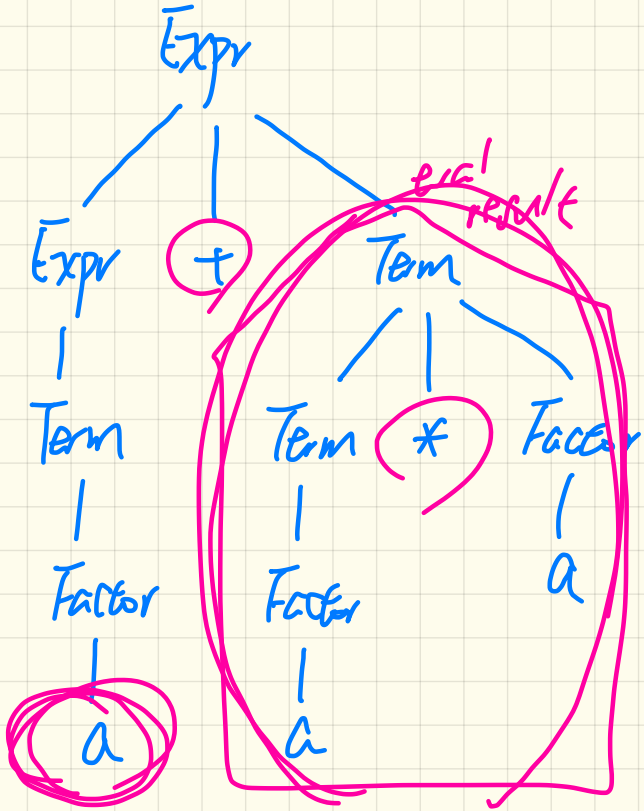
stack: s0 s1 s2 s2 s2 bad

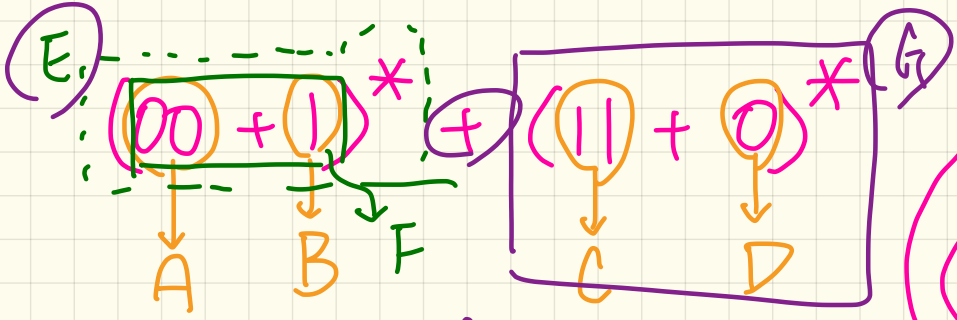
Expr	→	*Expr* + *Term*
	|	*Term*
Term	→	*Term* * *Factor*
	|	*Factor*
Factor	→	(*Expr*)
	|	a

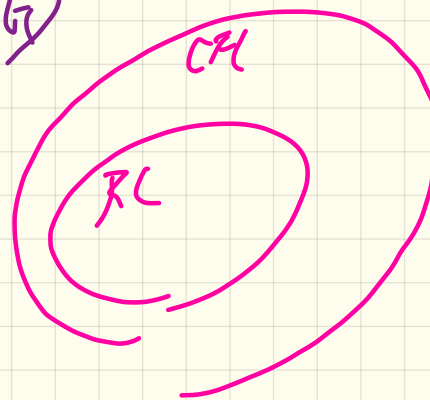
a + a * a

evaluation

post-order







$S \rightarrow E \mid G$

$E \rightarrow \varepsilon \mid FE$

$F \rightarrow A \mid B$

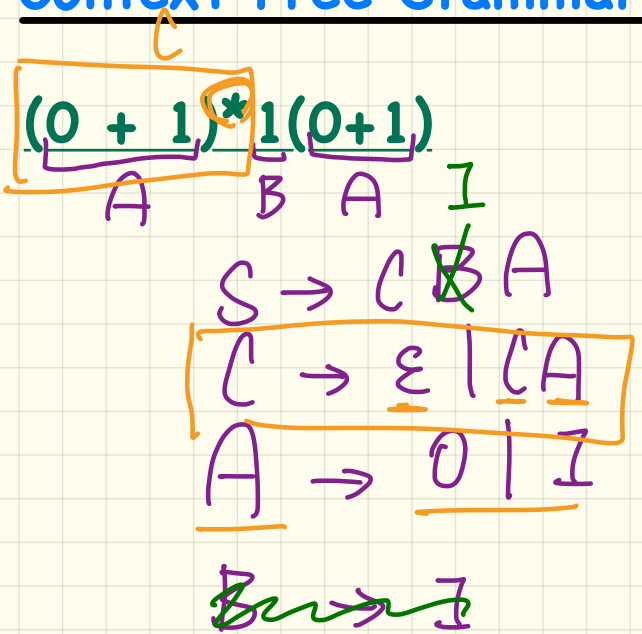
$A \rightarrow 00$

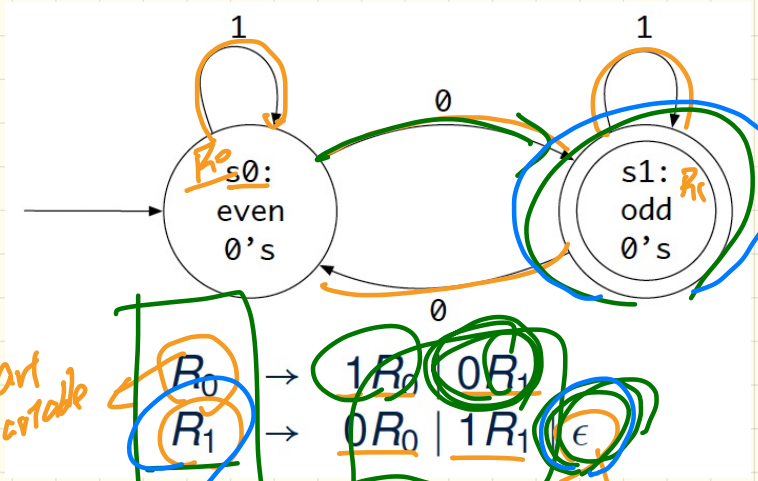
$B \rightarrow 1$

$C \rightarrow 11$

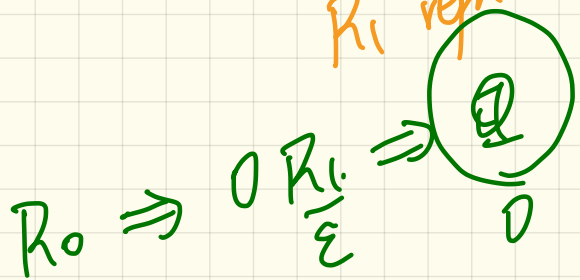
$D \rightarrow 0$

Context-Free Grammar (CFG): from RE

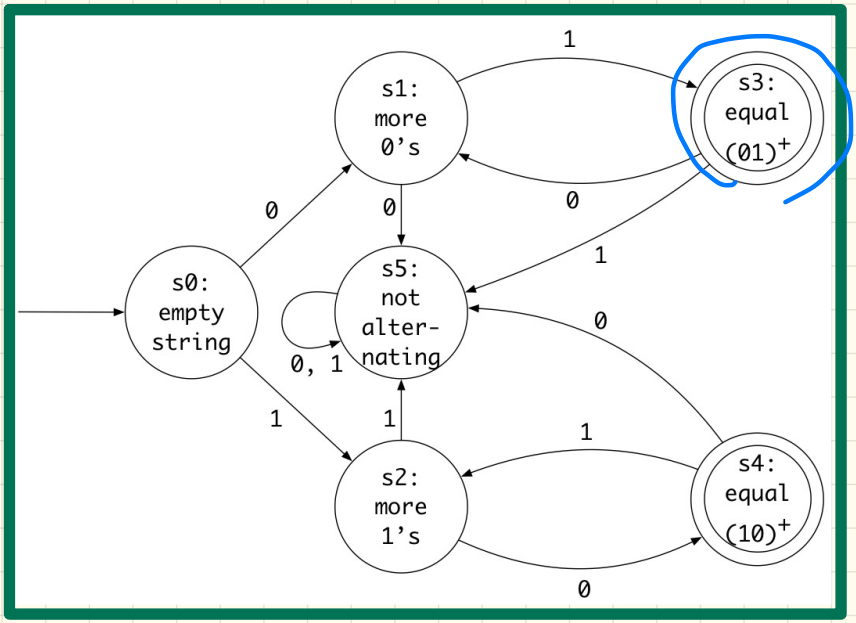




R_1 represents an accept state



Context-Free Grammar (CFG): from DFA



start variable

$S_0 \rightarrow$

$S_1 \rightarrow$

$S_2 \rightarrow$

$S_3 \rightarrow \varepsilon \mid 0S_1 \mid 1S_5$

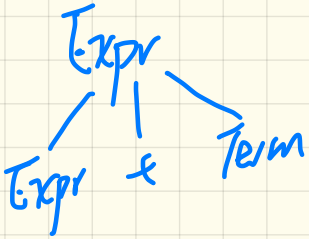
$S_4 \rightarrow \varepsilon$

$S_5 \rightarrow 0S_4 \mid 1S_3$

Context-Free Grammar (CFG): Leftmost Derivation

Expr	→	*Expr* + *Term*
	|	*Term*
Term	→	*Term* * *Factor*
	|	*Factor*
Factor	→	(*Expr*)
	|	a

Parse Tree: a + a * a



Derivation: a + a * a

Expr ⇒ Expr + Term

⇒ Term + Term

⇒ Factor + Term

⇒ a + Term

⇒ a + Term * Factor

⇒ a + Factor * Factor

⇒ a + a * Factor

⇒ a + a * a

Context-Free Grammar (CFG): Rightmost Derivation

Expr	→	*Expr* + *Term*
	|	*Term*
Term	→	*Term* * *Factor*
	|	*Factor*
Factor	→	(*Expr*)
	|	a

Derivation: a + a * a

Parse Tree: a + a * a

Context-Free Grammar (CFG): Leftmost Derivation

Expr	→	*Expr* + *Term*
	|	*Term*
Term	→	*Term* * *Factor*
	|	*Factor*
Factor	→	(*Expr*)
	|	a

Derivation: (a + a) * a

Parse Tree: a + a * a

Context-Free Grammar (CFG): Rightmost Derivation

Expr	→	*Expr* + *Term*
	|	*Term*
Term	→	*Term* * *Factor*
	|	*Factor*
Factor	→	(*Expr*)
	|	a

Derivation: (a + a) * a

Parse Tree: a + a * a

Given input string $w \in \Sigma^*$

derivation 1

$\neq$

derivation 2 for w

$\Downarrow$

G is ambiguous.

$$(\boxed{a} + \boxed{a}) * \boxed{a}$$

Exercise

ex1. $a + a * a$

ex2. $a + (a * a)$

ex3. $(a + a) * a$

→ meaning 1

→ meaning 2.

do they share the same meaning according to

Expr	→	*Expr* + *Term*
	|	*Term*
Term	→	*Term* * *Factor*
	|	*Factor*
Factor	→	(*Expr*)
	|	a

ambiguous

a * a * a

Expr

Term

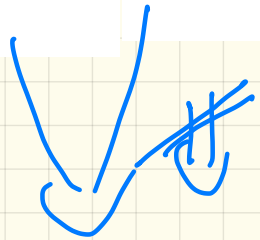
Term * Term

Unique leftmost derivation for the string `(a + a) * a`:

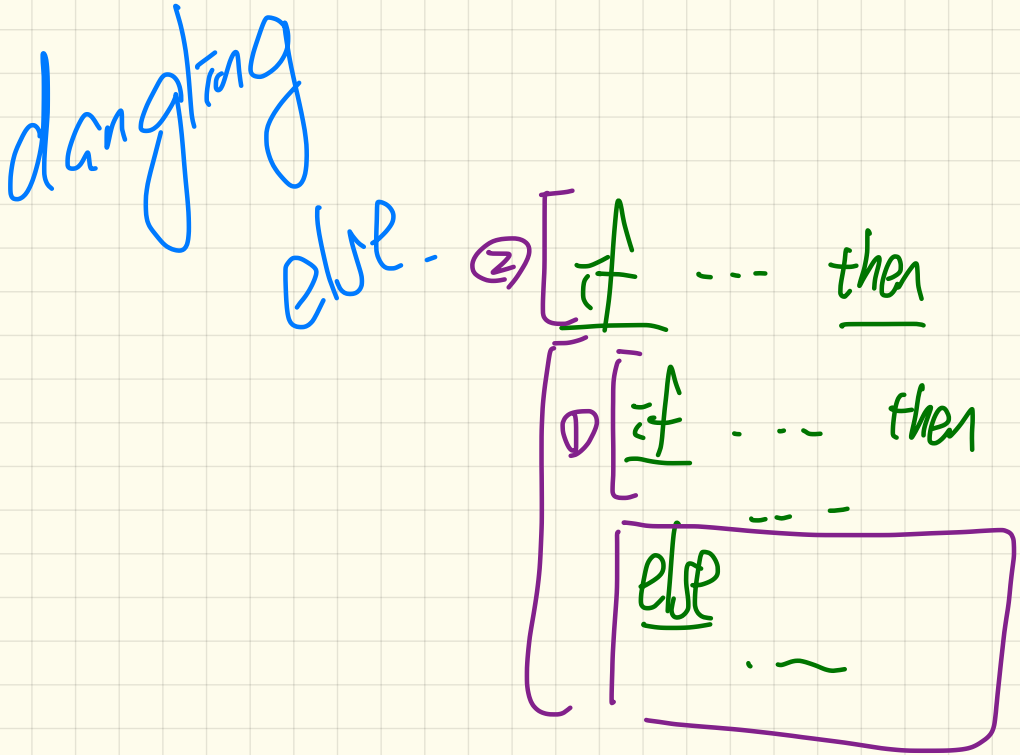
$$
\begin{aligned}
\mathit{Expr} &\Rightarrow \mathit{Term} \\
&\Rightarrow \mathit{Term} * \mathit{Factor} \\
&\Rightarrow \mathit{Factor} * \mathit{Factor} \\
&\Rightarrow (\ \mathit{Expr}\ ) * \mathit{Factor} \\
&\Rightarrow (\ \mathit{Expr} + \mathit{Term}\ ) * \mathit{Factor} \\
&\Rightarrow (\ \mathit{Term} + \mathit{Term}\ ) * \mathit{Factor} \\
&\Rightarrow (\ \mathit{Factor} + \mathit{Term}\ ) * \mathit{Factor} \\
&\Rightarrow (\ a + \mathit{Term}\ ) * \mathit{Factor} \\
&\Rightarrow (\ a + \mathit{Factor}\ ) * \mathit{Factor} \\
&\Rightarrow (\ a + a\ ) * \mathit{Factor} \\
&\Rightarrow (\ a + a\ ) * a
\end{aligned}
$$

Unique rightmost derivation for the string `(a + a) * a`:

$$
\begin{aligned}
\mathit{Expr} &\Rightarrow \mathit{Term} \\
&\Rightarrow \mathit{Term} * \mathit{Factor} \\
&\Rightarrow \mathit{Term} * a \\
&\Rightarrow \mathit{Factor} * a \\
&\Rightarrow (\ \mathit{Expr}\ ) * a \\
&\Rightarrow (\ \mathit{Expr} + \mathit{Term}\ ) * a \\
&\Rightarrow (\ \mathit{Expr} + \mathit{Factor}\ ) * a \\
&\Rightarrow (\ \mathit{Expr} + a\ ) * a \\
&\Rightarrow (\ \mathit{Term} + a\ ) * a \\
&\Rightarrow (\ \mathit{Factor} + a\ ) * a \\
&\Rightarrow (\ a + a\ ) * a
\end{aligned}
$$



G is ambiguous
∵ the two derivations not done in the same order.


dangling
else
if ... then
if ... then
else