AUTOMATA FORMAL LANGUAGES AND LOGIC

Lecture notes on Context Free Grammar/Linear Grammar

Prepared by:
Prof.Sangeeta V I
Assistant Professor

Department of Computer Science & Engineering

PES UNIVERSITY

(Established under Karnataka Act No.16 of 2013)
100-ft Ring Road, BSK III Stage, Bangalore – 560 085

Table of contents

Examples Solved

1. **Context Free Grammar (CFG)** is defined by 4 tuples **G=(V,T,P,S)** where,

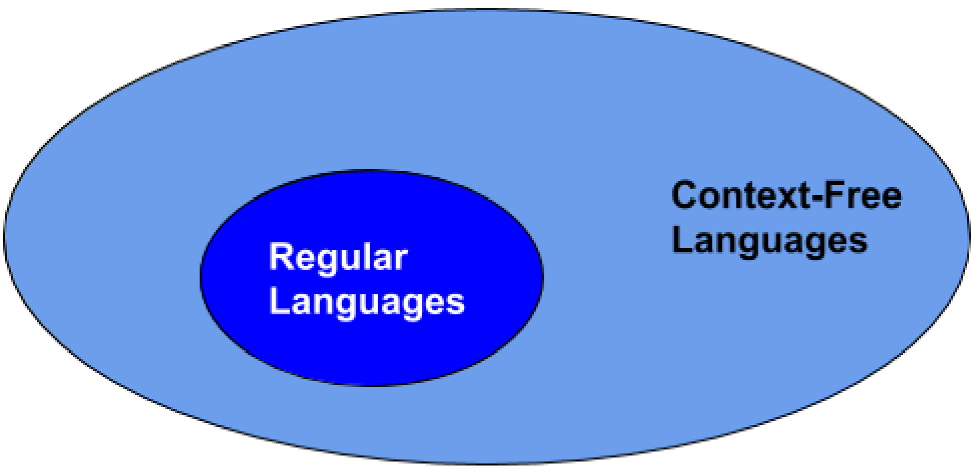
V is the set of variables.
T is the set of terminal symbols.
P is the set of production rules of the form:
A→α (Variable→String) , Where $\alpha=\{V \cup \Sigma\}^*$ and $A \in V$.
A context-free grammar has no restrictions on the right side of its productions, while the left side must be a single variable.

S is the start symbol.

- CFG may not be right-linear or left-linear, i.e., it may have a variable in the middle of the right-hand side of the grammar, surrounded by terminal symbols on both sides.
- CFG may not even be linear, i.e., may have more than one variable on the right-hand side.
- If G is a CFG with alphabet Σ and start symbol S, then the language of G is the set $L(G)=\{ w \in \Sigma^* \mid S \Rightarrow^* w \}$.
- Any language generated by a context free grammar(CFG)is a Context Free Language(CFL).
- CFG / Linear grammar has only one variable on the RHS of any production rule.

2. **Context-Free Languages**

Context-free languages are a strict superset of the regular languages. Every regular language is context-free, but not necessarily the other way around.



Example 1:

Construct linear grammar for the even palindromes. $L=\{ww^R, w\in\{a,b\}^*\}$

Palindrome is a sequence that reads the same backwards as forwards, e.g. madam.
Set of strings that belong to the language $=\{\lambda, aa, bb, abba, \ldots\ldots\}$
Minimum strings:λ
$w\ w^R=a\ a$
$w\ w^R=b\ b$
$w\ w^R=ab\ ba$ and so on.

There is a pattern which follows,
if we generate an ‘a’ at the start there has to be an ‘a’ at the end.

S→**aSa**

Similarly,if we generate ‘b’ at the start there has to be ‘b’ at the end.

S→aSa|**bSb**

Since it is an even palindrome,we will replace the S with λ.

CFG/linear grammar for the even palindromes is:

S→aSa|bSb|λ

Example 2:

Construct linear grammar for the even palindromes. $L=\{w\mathbf{C}w^R, w\in\{a,b\}^*\}$

C represents the mid.
We can take the grammar from the previous example and replace S→λ with S→C.

CFG/linear grammar for the even palindromes with C as seperator is:

S→aSa|bSb|C

Example 3:

Construct linear grammar for $L=\{ww^R, w\in\{ab\}^*|(ba)^*\}$

w here is either ab or ba.

If we generate 'ab' at the start we must generate 'ba' at the end.
S→**ab**S**ba**

If we generate 'ba' at the start we must generate 'ab' at the end.
S→**ba**S**ab**

CFG/linear grammar for $L=\{ww^R, w\in\{ab\}^*|(ba)^*\}$ is:

S→abSba|baSab|λ

Example 4:

Construct linear grammar for $L=\{a^n ww^R b^n \mid w\in\{a,b\}^*\}$

We will first generate $a^n b^n$.
S→**aSb**

Next,we will replace S with ww^R
Introduce new production S→**A,**where A will take care of ww^R.
S→aSb | **A**
A→aAa|bAb|λ

CFG/linear grammar for $L=\{a^n ww^R b^n \mid w\in\{a,b\}^*\}$ is:

S→aSb | A
A→aAa|bAb|λ

Example 5:

Construct linear grammar for $L=\{a^n b^{n+1}, n>=0\}$

$a^n b^{n+1}$ is similar to $a^n b^n$ but has an extra b.

$a^n b^{n+1} = a^n b b^n$

CFG/linear grammar for $L=\{a^n b^{n+1} \mid n>=0\}$ is:

S→aSb |b

Example 6:

Construct linear grammar for $L=\{a^{n+2}b^n, n>=1\}$

$a^{n+2}b^n=a^na^2b^n$

Minimum value n=1 , aa^2b=aaab

n=2 ,$a^2a^2b^2$=aaaabb

n=3 ,$a^3a^2b^3$=aaaaabbb and so on

There is a pattern which follows,
there are an equal number of a's and b's in the middle there are two a's.

CFG/linear grammar for $L=\{a^{n+2}b^n, n>=1\}$ is:

S→aSb |aaab

Example 7:

Construct linear grammar for $L=\{a^nb^{2n}, n>=0\}$

Enumerate the strings in L:

n	w(string)
0	λ
1	a bb
2	aa bbbb

There is a pattern which follows, for every 'a' there are two b's.
S→aSbb |λ

CFG/linear grammar for $L=\{a^nb^{2n}, n>=0\}$ is:

S→aSbb |λ

Example 8:

Construct linear grammar for $L=\{a^n b^{n-3}, n>=3\}$

Enumerate the strings in L:

n	w(string)
3	a^3b^0
4	$a^4b^{4-3}=a^4b^1 = aa^3b$
5	$a^5b^{5-3}=a^5b^2= a^2a^3b^2$

There is a pattern which follows,
there are an equal number of a's and b's and extra three a's.

S→aSb |aaa

CFG/linear grammar for $L=\{a^n b^{n-3}, n>=3\}$ is:

S→aSb |aaa

Example 9:

Construct linear grammar for $L=\{a^n b^m, n>m\}$

Strings in the language L={more a's than b's}={a,aa…,aab,aaaabbbb, ...}

The pattern which follows is at least one more 'a' along with the same number of a's and b's.

S→aSb |aS|a (S→**aS**|**a** will generate one or more a's)

CFG/linear grammar for $L=\{a^n b^m, n>m\}$ is:

S→aSb |aS|a

Example 10:

Construct linear grammar for $L=\{a^n b^m, n \neq m\}$

$a^n b^m, n \neq m$ means either the number of a's are more or number of b's are more.

S→A | B (A-takes care of more a's , B-takes care of more b's)

A→aAb|aA|a

B→aBb|bB|b

CFG/linear grammar for $L=\{a^n b^m, n \neq m\}$ is:

S→A | B
A→aAb|aA|a
B→aBb|bB|b

Example 11:

Construct linear grammar for $L=\{a^n b^m, n=2+(m \bmod 3)\}$

Enumerate the strings in L:

m	n=2+(m mod 3)	$w=a^n b^m$
0	n=2+0mod3=2+0=2	aa
1	n=2+1mod3=2+1=3	aaab
2	n=2+2mod3=2+2=4	aaaabb
3	n=2+3mod3=2+0=2	aabbb
4	n=2+4mod3=2+1=3	aaabbbb
5	n=2+5mod3=2+2=4	aaaabbbb

We see that there is a restriction on the number of a's=2,3 or 4.
The number of b's are always multiples of 3.

Base cases: aa**bbb**,aaab**bbb**,aaaabb**bbb**

S→aa**A**|aaab**A**|aaaabb**A**
A→**bbbA**|λ

CFG/linear grammar for $L=\{a^n b^m, n=2+(m \bmod 3)\}$ is:

S→aaA|aaabA|aaaabbA
A→bbbA|λ

Example 12:

Construct linear grammar for $L=\{a^n b^m, n \neq 2m\}$

L={number of a's ≠ twice the number of b's}

aab, aaaabb, aaaaaabbb ,......∉ L

Enumerate the strings in L:

m (#b's)	**n (#a's)**	**w , n≠2m**
0	≠0	No b's, At least one 'a'=a^+
1	≠2	One 'b'. No of a's ≠2 w∈{b,ab,aaab,aaaab,aaaaab.....}
2	≠4	Two b's No of a's ≠4 w∈{bb,abb,aabb,aaabb,aaaaabb,aaaaaabb,......}
3	≠6	Three b's No of a's ≠6 w∈{bbb,abbb,aabbb,aaabbb,aaaaabbb,aaaaaabbb,......}
4	≠8	Four b's No of a's ≠8 w∈{bbbb,abbbb,aabbbb,aaabbbb,aaaaabbbb,aaaaaabbbb,......}

Strings in the language :
{**b**,ab,aaab,aaaab,aaaaab.....}
{**bb**,abb,aabb,aaabb,aaaaabb,aaaaaabb,.....}
{**bbb**,abbb,aabbb,aaabbb,aaaaabbb,aaaaaaabbb,......}
{**bbbb**,abbbb,aabbbb,aaabbbb,aaaaabbbb,aaaaaabbbb,......}
General form :
{$\mathbf{b}^+$,....}

{b,**ab**,aaab,aaaab,aaaaab.....}
{bb,**abb**,aabb,aaabb,aaaaabb,aaaaaabb,.....}
{bbb,**abbb**,aabbb,aaabbb,aaaaabbb,aaaaaaabbb,......}
{bbbb,**abbbb**,aabbbb,aaabbbb,aaaaabbbb,aaaaaabbbb,......}
General form :
{b^+,**ab***,....}

{b,ab,**aab**,aaab,aaaab,aaaaab.....}
{bb,abb,**aabb**,aaabb,aaaaabb,aaaaaabb,.....}
{bbb,abbb,**aabbb**,aaabbb,aaaaabbb,aaaaaaabbb,......}
{bbbb,abbbb,**aabbbb**,aaabbbb,aaaaabbbb,aaaaaabbbb,......}
Pattern is,aa(b^+)b
General form :
{b^+,ab*,**aab^+b**,....}

{b,ab,aab,**aaab**,aaaab,aaaaab.....}
{bb,abb,aabb,**aaabb**,aaaaabb,aaaaaabb,.....}
{bbb,abbb,aabbb,**aaabbb**,aaaaabbb,aaaaaaabbb,......}
{bbbb,abbbb,aabbbb,**aaabbbb**,aaaaabbbb,aaaaaabbbb,......}
Pattern is,aa(b^+)b
General form :
{b^+,ab*,aab^+b,**aaab*b**,....}

aab, **aa**aab**b**, **aa**aaaabb**b** ,......∉ L

We do not want to generate **'aab'** ,so we build upon it so it doesn't occur.

The cases to handle which we saw so far:

1. At Least one a = a^+
2. b^+
3. ab*

CFG/linear grammar for L={$a^n b^m$,n≠2m}is:

S→aaSb|A|B|aC

A→aA|a (case 1)

B→Bb|b (case 2)

C→Cb|λ (case 3)

Example 13:

Construct linear grammar for $L=\{a^{n+2}b^m, m>n, n>=0\}$

n (#a's)	m (#b's)	String
0	>0	$a^{n+2}b^m=a^2b^m=a^2bb*$
1	>1	$a^{n+2}b^m=a^{1+2}b^m=aa^2bbb*$
2	>2	$a^{n+2}b^m=a^{2+2}b^m=aaa^2bbbb*$

Strings to understand the pattern:

a^2bb*

aa^{2}bbb*

aaa^{2}bbbb*

The cases to handle which we saw so far:

1. Same number of a's and b's.
2. a^2b=aab ,substring (in the middle).
3. b*any number of b's at the end.

S→ aSb (case 1)

S→ aSb|aab (case 2)

S→ aSb|aab|Sb (case 3)

CFG/linear grammar for $L=\{a^{n+2}b^m, m>n, n>=0\}$ is:

S→ aSb|aab|Sb

Example 14:

Construct linear grammar for $L=\{a^nb^mc^md^n, n,m>=1\}$

The cases to handle :

1. Equal number of a's and d's.
2. Equal number of b's and c's.

Minimum string in L=abcd

S→ aSd|aAd (case 1 and also n>=1, S→ aAd and not S→ A)
A→ bAc|bc **(**case 2,and also m>=1,A→bc and not A→λ **)**

CFG/linear grammar for $L=\{a^nb^mc^md^n, n,m>=1\}$ is:

S→ aSd|aAd
A→ bAc|bc

Example 15:

Construct linear grammar for $L=\{a^nb^mc^k, k=n+m, n,m,k>=0\}$

Number of c's=number of a's + number of b's

We can rewrite $a^nb^mc^k = a^nb^mc^{n+m} = a^nb^mc^mc^n$

The cases to handle (similar to the previous one):

1. Equal number of a's and c's.
2. Equal number of b's and c's.

CFG/linear grammar for $L=\{a^nb^mc^md^n, n,m>=1\}$ is:

S→ aSc|A (n,m>=0 ,λ is the minimum string in the language)
A→ bAc|λ

Example 16:

Construct linear grammar for $L=\{a^n b^m c^k, m=2n, k=2, n>=0\}$

We can rewrite $a^n b^m c^k = a^n b^{2n} c^2$

We can have two variables **A** to handle $\mathbf{a^n b^{2n}}$ and **B** to handle $\mathbf{c^2}$.

CFG/linear grammar forL=$\{a^n b^m c^k, m=2n, k=2, n>=0\}$ is:

S→ AB

A→ **aAbb|λ**
B→ cc

Example 17:

Construct linear grammar for $L=\{a^n b^m c^k, m,n>=0, k=n+2m\}$

We can rearrange rewrite $a^n b^m c^k = a^n b^m c^{n+2m} = a^n b^m c^{2m}.c^n$

The cases to handle :

3. Equal number of a's and c's.
4. Every 'b' has 'cc'.

CFG/linear grammar for $L=\{a^n b^m c^k, m,n>=0, k=n+2m\}$ is:

S→ aSc|A

A→bAcc|λ

Example 18:

Construct linear grammar for L={|w| mod 3 ≠ |w| mod 2,w∈{a}*}

Remainder for mod 3=0,1,2
Remainder for mod 3=0,1

We will find out a^n for which a^n mod 3≠ a^n mod 2

a^n	mod 2	mod 3
a^0	0	0
a^1	1	1
a^2	0	2
a^3	1	0
a^4	0	1
a^5	1	2
a^6	0	0
a^7	1	1
a^8	0	2
a^9	1	0
a^{10}	0	1
a^{11}	1	2

L={$\mathbf{a^2}$,$\mathbf{a^3}$,$\mathbf{a^4}$,$\mathbf{a^5}$,a^8,a^9,a^{10},a^{11},.......}
Base cases:$\mathbf{a^2}$,$\mathbf{a^3}$,$\mathbf{a^4}$,$\mathbf{a^5}$
Keep on adding 6 to each of $\mathbf{a^2}$,$\mathbf{a^3}$,$\mathbf{a^4}$,$\mathbf{a^5}$,we get $\mathbf{a^8}$,$\mathbf{a^9}$,$\mathbf{a^{10}}$,$\mathbf{a^{11}}$.
Generate S in multiples of 6.
S→aaaaaaS|aa|aaa|aaaa|aaaaa

CFG/linear grammar for L={|w| mod 3 ≠ |w| mod 2,w∈{a}*} is:

S→aaaaaaS|aa|aaa|aaaa|aaaaa