Pumping Lemma for CFL

The size of Parse Trees

Theorem 7.17: Suppose we have a parse tree according to a Chomsky-Normal-Form grammar $G = (V, T, P, S)$, and suppose that the yield of the tree is a terminal string w. If the length of the longest path is n, then $|w| \leq 2^{n-1}$.

PROOF: The proof is a simple induction on n.

The size of Parse Trees

Theorem 7.17: Suppose we have a parse tree according to a Chomsky-Normal-Form grammar $G = (V, T, P, S)$, and suppose that the yield of the tree is a terminal string w. If the length of the longest path is n, then $|w| \leq 2^{n-1}$.

PROOF: The proof is a simple induction on n.

- If $|w| > 2^{n-1}$, then the longest path is $> n$

The size of Parse Trees

Theorem 7.17: Suppose we have a parse tree according to a Chomsky-Normal-Form grammar $G = (V, T, P, S)$, and suppose that the yield of the tree is a terminal string w. If the length of the longest path is n, then $|w| \leq 2^{n-1}$.

PROOF: The proof is a simple induction on n.

- If $|w| > 2^{n-1}$, then the longest path is $> n$
- Let $|V| = m$

The size of Parse Trees

Theorem 7.17: Suppose we have a parse tree according to a Chomsky-Normal-Form grammar $G = (V, T, P, S)$, and suppose that the yield of the tree is a terminal string w. If the length of the longest path is n, then $|w| \leq 2^{n-1}$.

PROOF: The proof is a simple induction on n.

- If $|w| > 2^{n-1}$, then the longest path is $> n$
- Let $|V| = m$
- For $|w| = 2^m$, longest path is $> m + 1$

The size of Parse Trees

Theorem 7.17: Suppose we have a parse tree according to a Chomsky-Normal-Form grammar $G = (V, T, P, S)$, and suppose that the yield of the tree is a terminal string w. If the length of the longest path is n, then $|w| \leq 2^{n-1}$.

PROOF: The proof is a simple induction on n.

- If $|w| > 2^{n-1}$, then the longest path is $> n$
- Let $|V| = m$
- For $|w| = 2^m$, longest path is $> m + 1$
- In that longest path a variable must have been repeated (since we have only m variables).

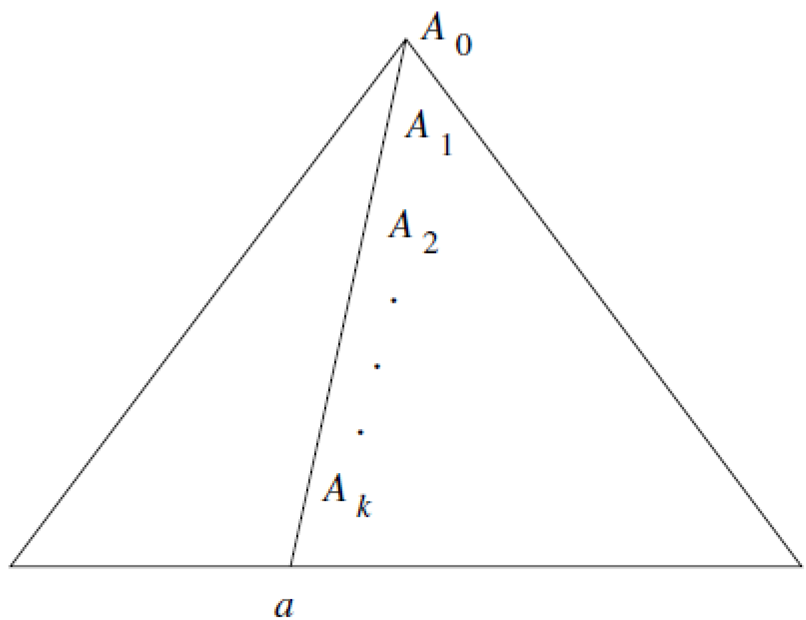


Figure 7.5: Every sufficiently long string in L must have a long path in its parse tree

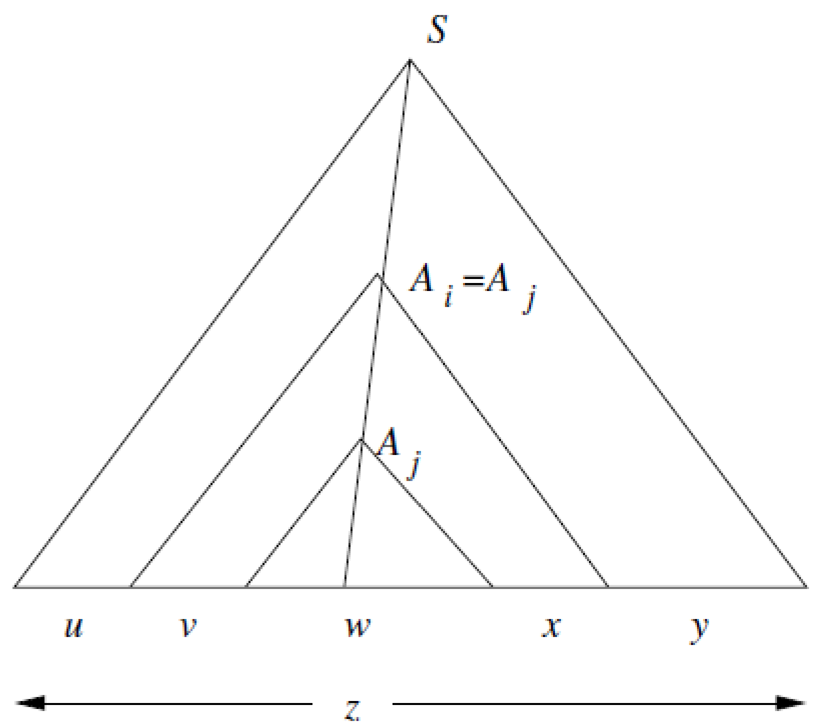
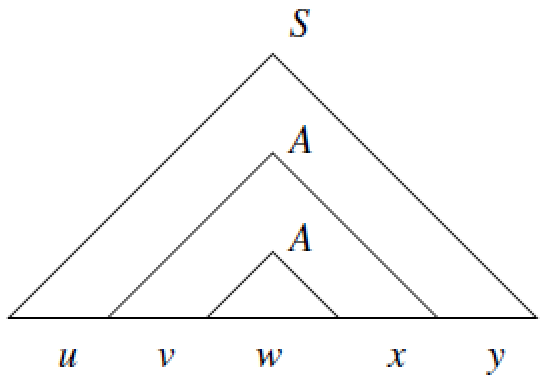
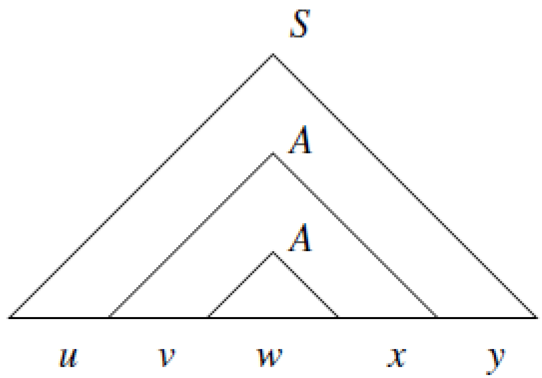
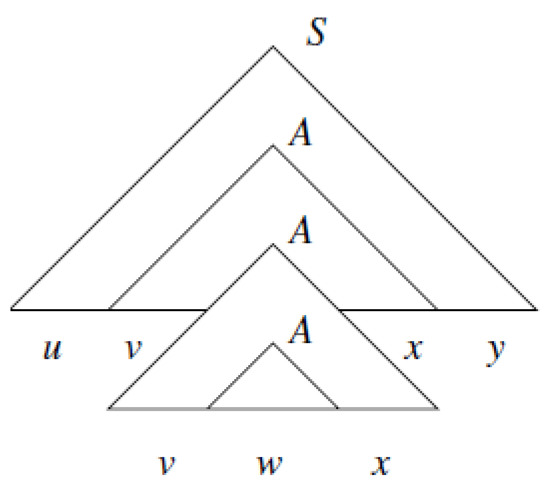
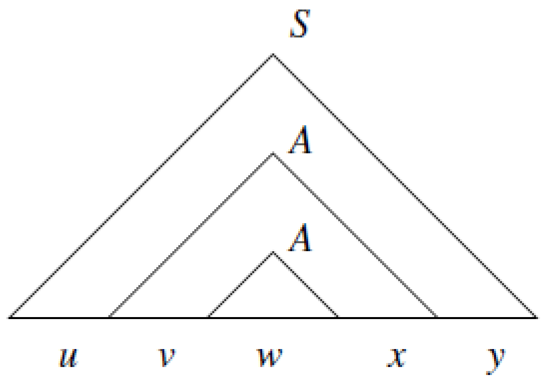


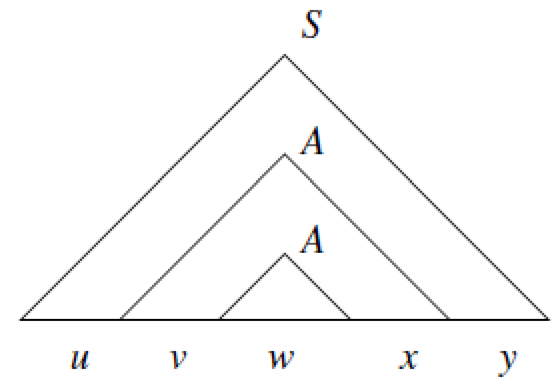
Figure 7.6: Dividing the string w so it can be pumped

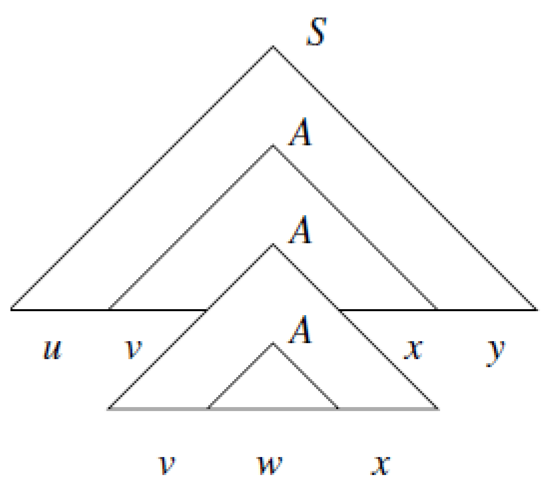

S
A
A
u
v
w
x
y

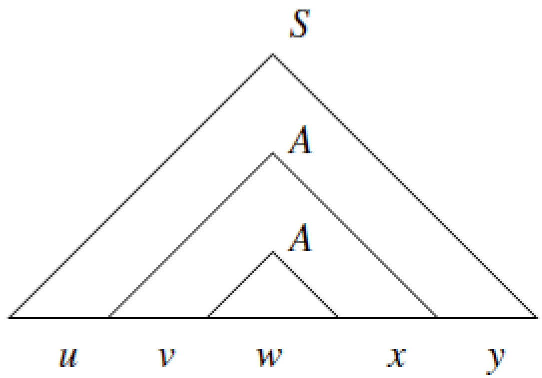

S
A
A
u v w x y

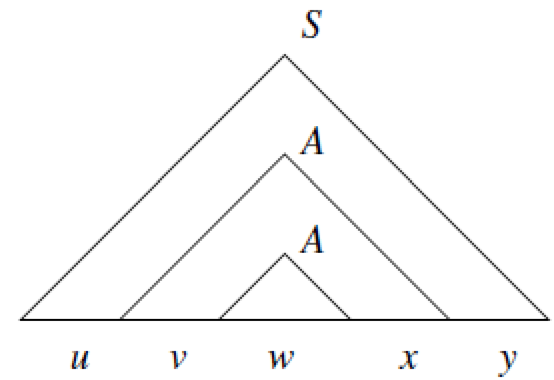

S
A
A
A
u v x y
v w x

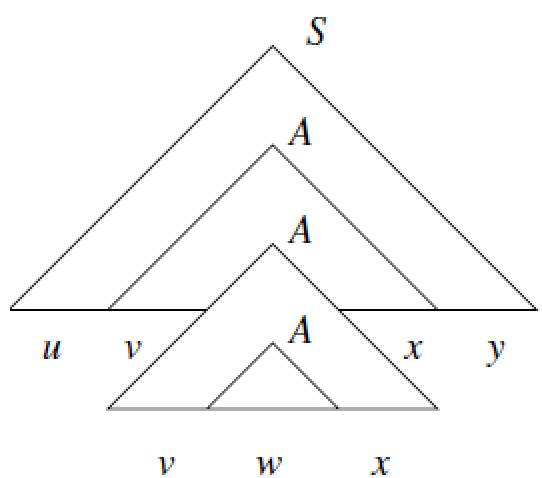

S
A
A
u v w x y

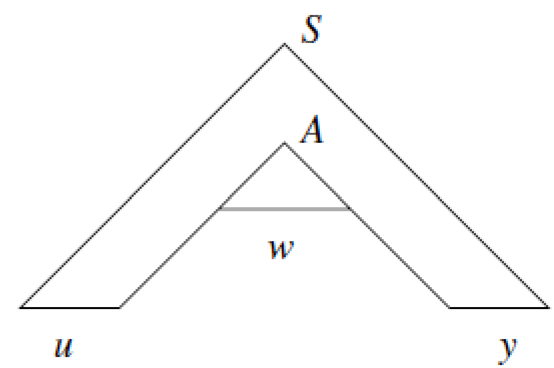

S
A
A
u v w x y


S
A
A
A
u v x y
v w x


S
A
A
u
v
w
x
y


S
A
A
u
v
w
x
y


S
A
A
A
u
v
x
y
v
w
x


S
A
w
u
y

Theorem 7.18: (The pumping lemma for context-free languages) Let L be a CFL. Then there exists a constant n such that if z is any string in L such that $|z|$ is at least n, then we can write $z = uvwxy$, subject to the following conditions:

1. $|vwx| \leq n$. That is, the middle portion is not too long.

2. $vx \neq \epsilon$. Since v and x are the pieces to be "pumped," this condition says that at least one of the strings we pump must not be empty.

3. For all $i \geq 0$, uv^iwx^iy is in L. That is, the two strings v and x may be "pumped" any number of times, including 0, and the resulting string will still be a member of L.

Example 7.19: Let L be the language $\{0^n1^n2^n \mid n \geq 1\}$. That is, L consists of all strings in $\mathbf{0}^+\mathbf{1}^+\mathbf{2}^+$ with an equal number of each symbol, e.g., 012, 001122, and so on. Suppose L were context-free. Then there is an integer n given to us by the pumping lemma. Let us pick $z = 0^n1^n2^n$.

Example 7.19: Let L be the language $\{0^n1^n2^n \mid n \geq 1\}$. That is, L consists of all strings in $\mathbf{0^+1^+2^+}$ with an equal number of each symbol, e.g., 012, 001122, and so on. Suppose L were context-free. Then there is an integer n given to us by the pumping lemma. Let us pick $z = 0^n1^n2^n$.

We can write $z = uvwxy$, where $|vwx| \leq n$ and v and x are not both ϵ.

Example 7.19: Let L be the language $\{0^n 1^n 2^n \mid n \geq 1\}$. That is, L consists of all strings in $\mathbf{0}^+\mathbf{1}^+\mathbf{2}^+$ with an equal number of each symbol, e.g., 012, 001122, and so on. Suppose L were context-free. Then there is an integer n given to us by the pumping lemma. Let us pick $z = 0^n 1^n 2^n$.

We can write $z = uvwxy$, where $|vwx| \leq n$ and v and x are not both ϵ.

Then we know that vwx cannot involve both 0's and 2's, since the last 0 and the first 2 are separated by $n+1$ positions.

Example 7.19 : Let L be the language $\{0^n1^n2^n \mid n \geq 1\}$. That is, L consists of all strings in $\mathbf{0^+1^+2^+}$ with an equal number of each symbol, e.g., 012, 001122, and so on. Suppose L were context-free. Then there is an integer n given to us by the pumping lemma. Let us pick $z = 0^n1^n2^n$.

We can write $z = uvwxy$, where $|vwx| \leq n$ and v and x are not both ϵ.

Then we know that vwx cannot involve both 0's and 2's, since the last 0 and the first 2 are separated by $n + 1$ positions.

It can have only 0s and 1s.

Or, it can have only 1s and 2s.

In both cases *i=2* can make the resulting string to have inequal number of 0s, 1s, and 2s.

Example 7.21 : Let $L = \{ww \mid w \text{ is in } \{0,1\}^*\}$.

- Show L is not a CFL.

- Note, $\{ww^R | w \in \{0,1\}^* \}$ is a CFL.
- How can you prove this??

Example 7.21: Let $L = \{ww \mid w \text{ is in } \{0,1\}^*\}$.

- Let pumping length is n.
- Let the string be $z = 0^n 1^n 0^n 1^n$
- z can be written as $uvwxy$, such that $|vwx| \leq n$ and $vx \neq \epsilon$
- There are 7 cases, based on where vwx can occur in z.
- In all these cases it can be shown that uwy is not in L.