10-0: **Fun with CFGs**

- Create a CFG for the language:
  - $L = \{0^n 1^n 2^n : n > 0\}$
  - $\{012, 001122, 000111222, 000011112222, \ldots\}$

10-1: **Fun with CFGs**

- $L = \{0^n 1^n 2^n : n > 0\}$ is not context-free!
- Why?
  - Need to keep track of how many 0's there are, and match 1's – and match 2's
  - Only one stack

10-2: **Non-Context-Free Languages**

- We will use a similar idea to the pumping lemma for regular languages to prove a language is not context-free
  - Regular Languages: if a string is long enough, there must be some state q that is repeated in the computation
  - Context-Free Languages: if a string is long enough, there must be some non-terminal A that is used twice in a derivation

10-3: **Repeating Non-Terminals**

If w is long enough, parse tree for w will have some non-terminal that repeats along a path from the root to some leaf

- Let $\phi(G)$ be the "fan out" of the grammar – the longest string that appears on the right-hand side of some rule
- Let height of a parse tree be the longest path from root to some leaf
- Longest possible string produced by a grammar G with height h is:

10-4: **Repeating Non-Terminals**

If w is long enough, parse tree for w will have some non-terminal that repeats along a path from the root to some leaf

- Let $\phi(G)$ be the "fan out" of the grammar – the longest string that appears on the right-hand side of some rule
- Let height of a parse tree be the longest path from root to some leaf
- Longest possible string produced by a grammar G with height h is: $\phi(G)^h$
- Smallest possible height for a parse tree produced by grammar G for a string of length n is:

10-5: **Repeating Non-Terminals**

If w is long enough, parse tree for w will have some non-terminal that repeats along a path from the root to some leaf

- Let $\phi(G)$ be the "fan out" of the grammar – the longest string that appears on the right-hand side of some rule
- Let height of a parse tree be the longest path from root to some leaf
- Longest possible string produced by a grammar G with height h is: $\phi(G)^h$

- Smallest possible height for a parse tree produced by grammar G for a string of length n is: $\log_{\phi(G)} n$

10-6: **Repeating Non-Terminals**

- Let $\phi(G)$ be the "fan out" of the grammar – the longest string that appears on the right-hand side of some rule
- Let height of a parse tree be the longest path from root to some leaf
- Longest possible string produced by a grammar G with height h is: $\phi(G)^h$
- Smallest possible height for a parse tree produced by grammar G for a string of length n is: $\log_{\phi(G)} n$

Given any grammar G and integer k, there exists an n such that any string of length n must have a height $\geq k$

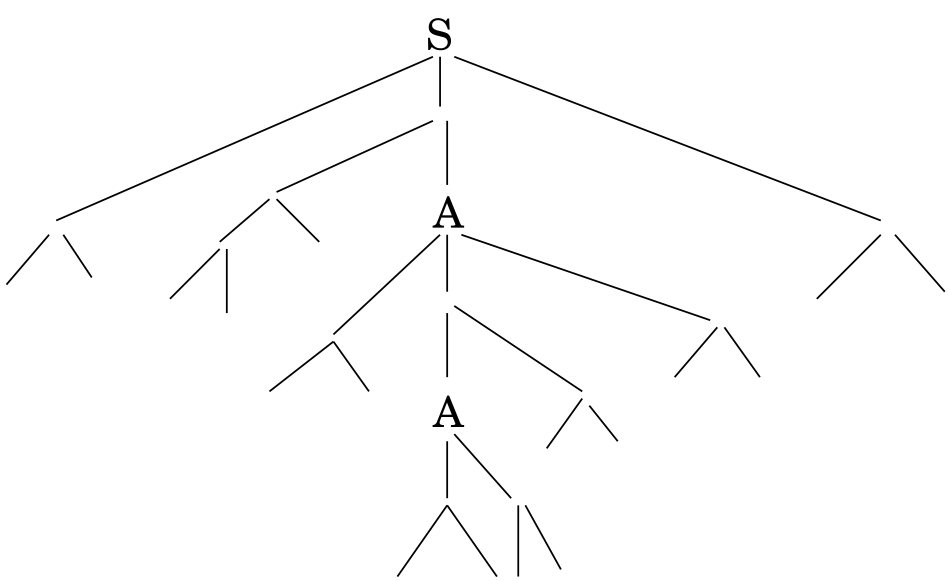
10-7: **Repeating Non-Terminals**

- Given any grammar G and integer k, there exists an n such that any string of length n must have a parse tree of height $\geq k$
- If a parse tree has a height $\geq k$, and the number of non-terminals in the string is $< k$ then by the pigeonhole principle ...

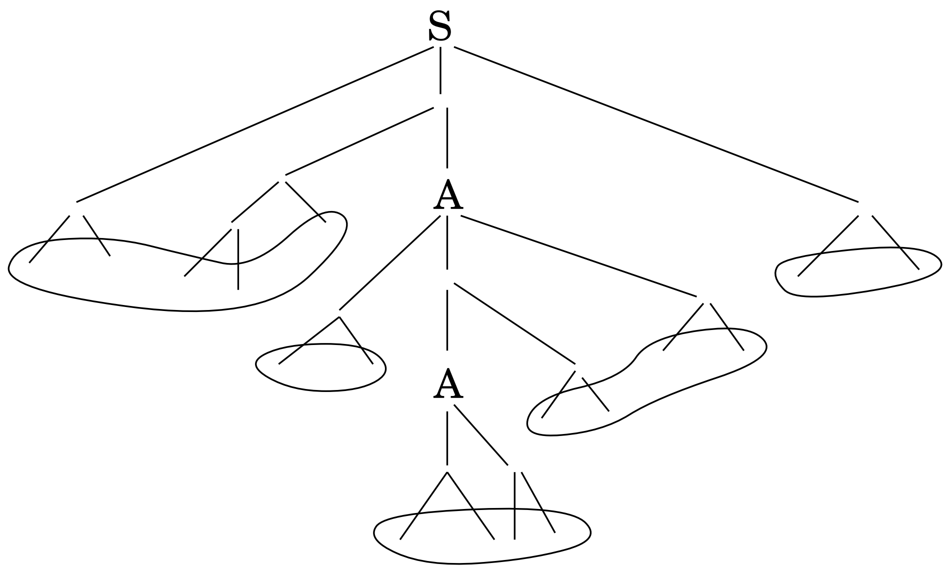
10-8: **Repeating Non-Terminals**

- Given any grammar G and integer k, there exists an n such that any string of length n must have a parse tree of height $\geq k$
- If a parse tree has a height $\geq k$, and the number of non-terminals in the string is $< k$ then by the pigeonhole principle, some non-terminal must repeat along a path from the root to some leaf.

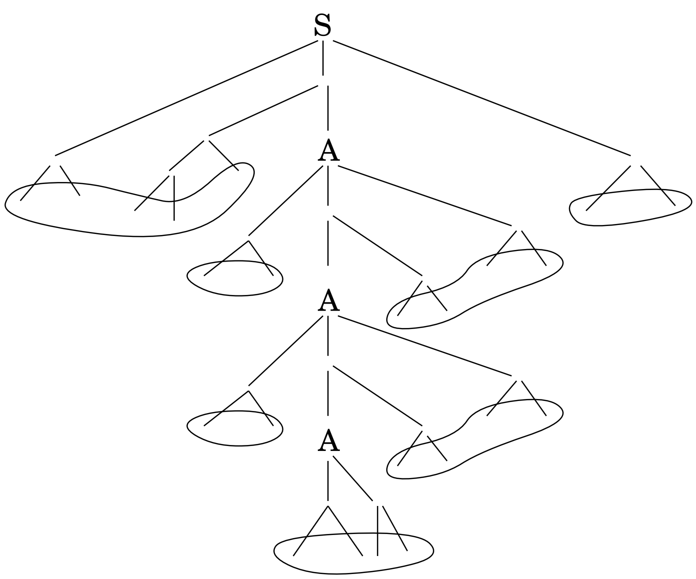
10-9: **Repeating Non-Terminals**



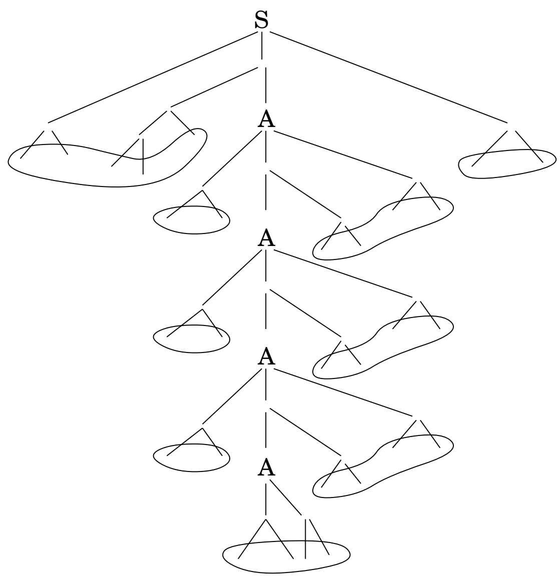
10-10: **Repeating Non-Terminals**



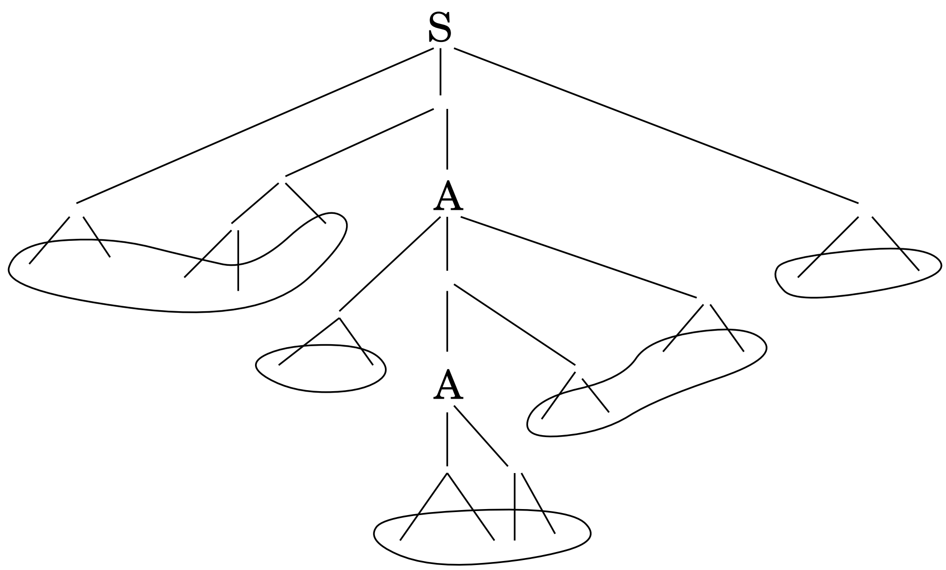
10-11: **Repeating Non-Terminals**



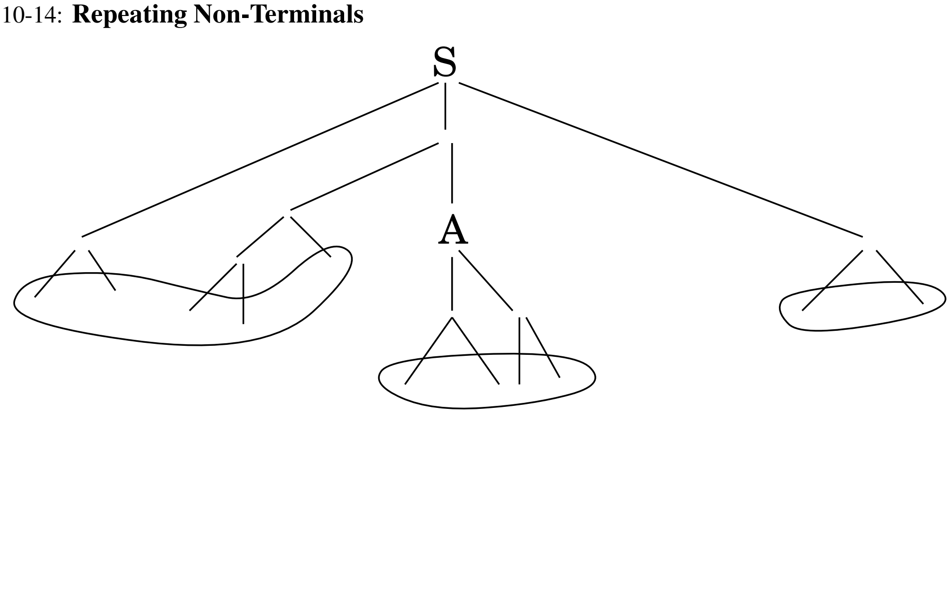
10-12: **Repeating Non-Terminals**



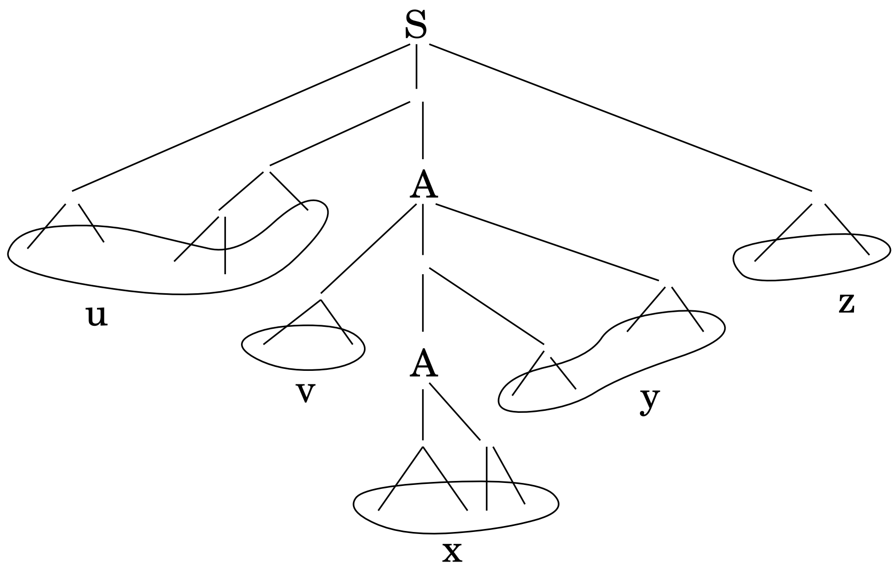
10-13: **Repeating Non-Terminals**



10-14: **Repeating Non-Terminals**



10-15: **Repeating Non-Terminals**



10-16: **CF Pumping Lemma**

- Given any Context-Free language L, there exists an integer n, such that for all $w \in L$ with $|w| \geq n$ can be broken into $w = uvxyz$ such that
  - $|vy| > 0$
  - $uv^ixy^iz \in L$ for all $i \geq 0$

10-17: **CF Pumping Lemma**
$L = \{0^n1^n2^n : n > 0\}$ is not Context Free

- Let n be the constant of the Context-Free Pumping lemma
- Consider $w = 0^n1^n2^n$
- If we break $w = uvxyz$, there are 4 cases:
  - v contains both 0's and 1's, or both 1's and 2's, or y contains both 0's and 1's, or both 1's and 2's
  - Neither v nor y contain any 0's
  - Neither v nor y contain any 1's
  - Neither v nor y contain any 2's

10-18: **CF Pumping Lemma**

- v contains both 0's and 1's, or both 1's and 2's, or y contains both 0's and 1's, or both 1's and 2's
  uv^2xy^2z is not in $0^*1^*2^*$, and is not in L
- Neither v nor y contain any 0's
  uv^2xy^2z contains either more 1's than 0's, or more 2's than 0's, and is not in L
- Neither v nor y contain any 1's
  uv^2xy^2z contains either more 0's than 1's, or more 2's than 1's, and is not in L

- Neither v nor y contain any 2's
  uv^2xy^2z contains either more 1's than 2's, or more 0's than 2's, and is not in L

10-19: **CF Pumping Lemma**

$L = \{a^n : n \text{ is prime }\}$ is not Context Free

10-20: **CF Pumping Lemma**

$L = \{a^n : n \text{ is prime }\}$ is not Context Free

- Let n be the constant of the Context-Free Pumping lemma
- Consider $w = a^p$, where p is the smallest prime number $> n$.
- If we break $w = uvxyz$:
  - Let $|vy| = k$, $|uxz| = r = n - k$
  - $uv^ixy^iz = a^{r+ik}$, $r + ik$ must be prime for all i
  - set $i = r + k + 1$: $r + ik = r + kr + k^2 + k = (r + k)(k + 1)$
  - uv^ixy^iz is not in L for $i = r + k + 1$; L is not Context-Free

10-21: **CF Closure Properties**

- Are the Context-Free Languages closed under union?

10-22: **CF Closure Properties**

- Are the Context-Free Languages closed under union?
  - YES!
  - Proved in Lecture 8 (when showing $L_{REG} \subseteq L_{CFG}$)
- Are the Context-Free Languages closed under intersection?

10-23: **CF Closure Properties**

- Are the Context-Free Languages closed under union?
  - YES!
  - Proved in Lecture 8 (when showing $L_{REG} \subseteq L_{CFG}$)
- Are the Context-Free Languages closed under intersection?
  - Hint – can we intersect two Context-Free langauges to get $0^n1^n2^n$?

10-24: **CF Closure Properties**

- Are the Context-Free Languages closed under intersection?

$L_1 = 0^n1^n2^*$	$L_2 = 0^*1^n2^n$
$S_1 \rightarrow A_1B_1$	$S_2 \rightarrow A_2B_2$
$A_1 \rightarrow 0A_11\|01$	$A_2 \rightarrow 0A\|\epsilon$
$B_1 \rightarrow 2B_1\|\epsilon$	$B_2 \rightarrow 1B_22\|12$

- $L_1 \cap L_2 = 0^n1^n2^n$
- L_1 is Context-Free, L_2 is Context-Free, $L_1 \cap L_2$ is not Context-Free

10-25: **CF Closure Properties**

- The Context-Free Languages are not closed under intersection
  - What if we tried to use the machine construction proof that showed that L_{DFA} is closed under intersection – why wouldn't that work?

10-26: **CF Closure Properties**

- Are the Context-Free Languages closed under intersection with a regular language?
  - That is, if L_1 is Context-Free, and L_2 is regular, must $L_1 \cap L_2$ be Context-Free?

10-27: **CF Closure Properties**

- Are the Context-Free Languages closed under intersection with a regular language?
  - That is, if L_1 is Context-Free, and L_2 is regular, must $L_1 \cap L_2$ be Context-Free?
- Run PDA L_1 and DFA L_2 "in parallel" (just like the intersection of two regular languages)

10-28: **CF Closure Properties**
$M_1 = (K_1, \Sigma, \Gamma_1, \Delta_1, s_1, F_1) \qquad M_2 = (K_2, \Sigma, \delta_2, s_2, F_2)$

- $K = K_1 \times K_2$
- $\Gamma_1 = \Gamma$
- $s = (s_1, s_2)$
- $F = F_1 \times F_2$
- Δ:
  - For each transition in Δ of form $((q_1, a, \beta), (p_1, \gamma))$, for each state $q_2 \in K_2$, add $(((q_1, q_2), a, \beta), ((p_1, \delta(q_2)), \gamma))$ to Δ
  - For each transition in Δ of form $((q_1, \epsilon, \beta), (p_1, \gamma))$, for each state $q_2 \in K_2$, add $(((q_1, q_2), a, \beta), ((p_1, q_2), \gamma))$ to Δ

10-29: **CF Closure Properties**

- Is $L = \{(0 + 1 + 2)^*$: # of 0's = # of 1's = # of 2's $\}$ Context Free?

10-30: **CF Closure Properties**

- Is $L = \{(0 + 1 + 2)^*$: # of 0's = # of 1's = # of 2's $\}$ Context Free?
  - $L \cap 00^*11^*22^* = \{0^n1^n2^n : n > 0\}$ which is not Context Free
  - Context-Free language intersected with a regular language must be context free
  - L is not Context-Free

10-31: **CF Closure Properties**

- Is $L = \{www : w \in (a + b)^*\}$ Context Free?

10-32: **CF Closure Properties**

- Is $L = \{www : w \in (a+b)^*\}$ Context Free?
  - Intersect L with $a^*ba^*ba^*b$ to get L_1
  - $L_1 = a^nba^nba^nb$
  - If L_1 is not context-free, L is not context-free either

10-33: **CF Closure Properties**

- Is $L = \{a^nba^nba^nb : n \geq 0\}$ Context Free?

10-34: **CF Closure Properties**

- Is $L = \{a^nba^nba^nb : n \geq 0\}$ Context Free?
  - Let n be the constant of the context-free pumping lemma
  - Consider $w = a^nba^nba^nb$
  - If we break w into $uvxyz$, there are several possibilities:

10-35: **CF Closure Properties**

- If we break w into $uvxyz$, there are several possibilities:
  - Either v or y contains at least one b. Then $w' = uv^2xy^2z$ will contain more than 3 b's, and not be in L
  - Neither v nor y contains any characters from the first set of a's. In this case, $w' = uv^2xy^2z$ will be of the form $a^nba^mba^ob$, were either m or o is greater than n, and hence w' is not in L

10-36: **CF Closure Properties**

- If we break w into $uvxyz$, there are several possibilities:
  - Neither v nor y contains any characters from the second set of a's. In this case, $w' = uv^2xy^2z$ will be of the form $a^mba^nba^ob$, were either m or o is greater than n, and hence w' is not in L
  - Neither v nor y contains any characters from the third set of a's. In this case, $w' = uv^2xy^2z$ will be of the form $a^mba^oba^nb$, were either m or o is greater than n, and hence w' is not in L