Pumping Lemma for CFLs

- **Recall**: If L is a regular language, then all **sufficiently long** strings in L can be **pumped** to create new strings in L
  - *Key idea*: Finite states + Pigeonhole Principle ⇒ repeated state in sequence of visited states

- If L is a context-free language, then all **sufficiently long** strings in L can be **pumped** to create new strings in L
  - *Key idea*: Finite set of non-terminals + finitely many terminals in each rule + Pigeonhole Principle ⇒ repeated non-terminal in derivation of long strings

More formally…

- *Traditional version*: For any CFG $G = (N, A, S, P)$ there is an integer n such that all strings $w \in L(G)$ with $|w| \geq n$ have such a derivation:

  $S \Rightarrow^* u.X.v \Rightarrow^* u.x.X.y.v \Rightarrow^* u.x.z.y.v = w$ (where $|x.y| > 0$ and $|x.z.y| \leq n$)

- **Proof sketch**: There are finitely many rules, and each produces finitely many terminals. Hence, G can generate very long strings only with very deep parse trees, which must have some repeated non-terminal X on the deepest root-to-leaf path (by the Pigeonhole Principle)
- *Stronger version*: For any CFG $G = (N, A, S, P)$ there is an integer n such that $\forall k \geq 1$, all strings $w \in L(G)$ with $|w| \geq n^k$ have such a derivation:

  $S \Rightarrow^* u.X.v \Rightarrow^* u.x_1.X.y_1.v \Rightarrow^* u.x_1.x_2.X.y_2.y_1.v \Rightarrow^* \ldots \Rightarrow^*$
  $u.x_1.x_2.\ldots.x_k.X.y_k.\ldots.y_2.y_1.v \Rightarrow^* u.x_1.x_2.\ldots.x_k.z.y_k.\ldots.y_2.y_1.v = w$

  where each $|x_i.y_i| > 0$ and $|x_1 \cdots x_k.z.y_k \cdots y_1| \leq n^k$

Parikh's Theorem

- **Theorem [1961/1966]**: If concatenation ('.' operation) is commutative, then all context-free languages are regular.
  - The languages $\{a^n.b^n | n \geq 0\}$ and $\{(ab)^n | n \geq 0\}$ are "letter equivalent"
- **Corollary**: If $L \subseteq A^*$ and A is a singleton, then L is regular iff L is context-free
- Original proof involves a complicated rearrangement of parse trees
  - [J. ACM 1966 eds] "…among the most fundamental yet subtly difficult to prove in the theory [of context-free languages]"
  - [Lindqvist] "…it is remarkable that Parikh came up with the idea of the proof, since the exact conditions controlling the structures of the trees […] are non-trivial, in the sense that it is not obvious that those conditions must hold."

Simplified proof [Goldstine, 1977]

- $L = L(G)$ where $G = (N, A, S, P)$. Let n be the number from the strong PL.
- For every $U \subseteq N$ such that $S \in U$, let L_U be the subset of L that can be derived from S using exactly the non-terminals in U
- Clearly $L = \bigcup_{U \subseteq N} L_U$
- Define:

  $C = \{w \in L_U \mid |w| < n^{|U|}\}$ and

  $D = \{x.y \mid 0 < |x.y| \leq n^{|U|} \text{ and } X \Rightarrow^* x.X.y \text{ for some } X \in U\}$
- Note that both C and D are finite (and hence regular)
- We will show that L_U is letter equivalent to $C.D^*$

Proof (part 1, easy)

- Let $w \in C.D^*$. If $w \in C$ then $w \in L_U$
- Otherwise, $w = w_0.s$ where $w_0 \in C.D^*$ and $s \in D$ $(s \neq \varepsilon)$
- Hence $s = x.y$ where $X \Rightarrow^* x.X.y$ for some $X \in U$
- Since w_0 is shorter than w, by IH w_0 is letter-equivalent to some $w' \in L_U$
- Hence $S \Rightarrow^* w'$ by a derivation that includes every non-terminal in U, including X i.e., $S \Rightarrow^* u.X.v \Rightarrow^* u.z.v = w'$
- Hence $S \Rightarrow^* u.X.v \Rightarrow^* u.x.X.y.v$ which is letter-equivalent to $w'.x.y$, which in turn is letter-equivalent to $w_0.s = w$

Proof (part 2, tricky)

- Let $w \in L_U$. If $|w| < n^{|U|}$ then $w \in C \subseteq C.D^*$
- Else by the strong PL: $S \Rightarrow^*_{d_0} u.X.v \Rightarrow^*_{d_1} u.x_1.X.y_1.v \Rightarrow^*_{d_2} u.x_1.x_2.X.y_2.y_1.v \ldots$ $\Rightarrow^*_{d_{|U|}} u.x_1.x_2.\ldots.x_{|U|}.X.y_{|U|}.\ldots.y_2.y_1.v \Rightarrow^*_{d_{|U|+1}} w$

  where $X \in U$ and each $x_i.y_i \in D$
- *Crucial observation*: At least one of the sub-derivations $d_1, d_2, \ldots, d_{|U|}$ does not introduce any new non-terminals into the derivation
- Drop this sub-derivation d_i to generate a shorter string in L_U that is letter-equivalent to $w' \in C.D^*$ (IH)
- Now w and $w'.x_i.y_i$ are letter-equivalent and the latter is in $C.D^*$

Non-CFLs

- The language $\{a^{2^n} | n \geq 0\}$ is not a CFL
- The language $L_{\text{double}} = \{x.x \mid x \in \{a, b\}^*\}$ is not a CFL
- **Proof**: Suppose it is. Let n be the number from the (weak) PL and consider the string $w = 0^n.1^n.0^n.1^n \in L_{\text{double}}$
- Then $S \Rightarrow^* u.X.v \Rightarrow^* u.x.X.y.v \Rightarrow^* u.x.z.y.v = w$

  (where $|x.y| > 0$ and $|x.z.y| \leq n$)
- Argue based on whether x and y belong entirely within the same "block", entirely within adjacent "blocks", or if they span "blocks"
  - Repeat or eliminate a sub-derivation to generate a string $\notin L_{\text{double}}$

CFL closure under complement?

- Show that $\overline{L_{\text{double}}} = \{y \in \{a,b\}^* \mid \forall x \in \{a,b\}^*, y \neq x.x\}$ is a CFL

- **Observation 1**: Strings in $\overline{L_{\text{double}}}$ either have odd length or look like:

u	a	v	x	b	y

(or a and b swapped) where $|u| = |x|$ and $|v| = |y|$

- **Observation 2**: The above strings also look like:

u	a	x	v	b	y