

Theorem B. If $L = N(M_2)$ for some PDA M_2 with **one state**, then there (effectively) exists a context-free grammar G such that $L = L(G)$.

Proof.

Example. Consider the one state PDA

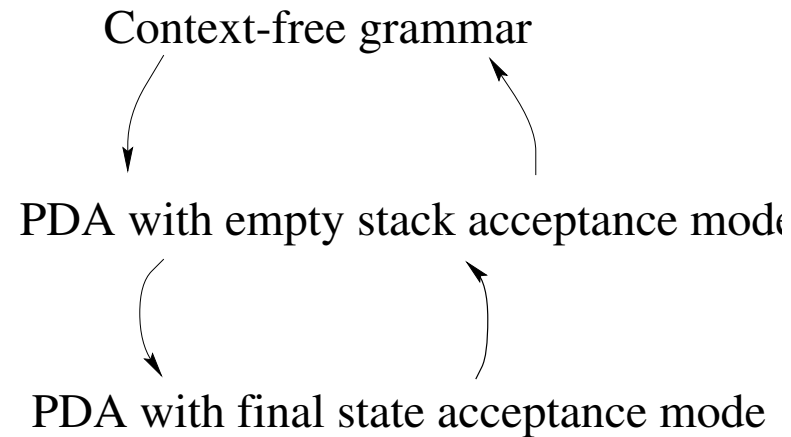
$$M = (\{\#\}, \{a, b\}, \{A, B, Z\}, \delta, \#, Z, \emptyset)$$

where δ contains transitions

$$\begin{aligned}\delta(\#, \varepsilon, Z) &= \{(\#, AZZA), (\#, B)\} \\ \delta(\#, a, A) &= \{(\#, \varepsilon)\} \\ \delta(\#, b, B) &= \{(\#, \varepsilon)\}\end{aligned}$$

Let us construct the equivalent grammar.

We have proved four theorems providing effective equivalence of three devices:



The constructions were all effective which means that when investigating closure properties of context-free languages or decision algorithms we may assume that the input is given in any of the three forms.

Pumping lemma for CFL

Not all languages are context-free. A technique for proving that some language is not context-free is a pumping lemma.

Just as in case of regular languages, pumping lemma states a property that every context-free language satisfies. If a language does not satisfy the pumping lemma then the language is not context-free.

Pumping lemma for CFL: Let L be a context-free language. Then there exists a positive number n such that every word $z \in L$ satisfying $|z| \geq n$ can be divided into **five** segments

$$z = uvwxy$$

in such a way that

$$\begin{cases} |vwx| \leq n, \text{ and} \\ v \neq \varepsilon \text{ or } x \neq \varepsilon \end{cases}$$

and for all $i \geq 0$ the word uv^iwx^iy is in the language L .

(The difference to the pumping lemma of regular languages is that now word z contains two subwords v and x that are pumped. Note that subwords v and x are always pumped the same number of times.)

Example. The language $L = \{a^m b^m \mid m \geq 0\}$ satisfies the pumping lemma.

Number $n = 2$ can be used. Let $z = a^m b^m$ be an arbitrary word of the language such that $|z| \geq 2$. This means $m \geq 1$. We break z into five parts as follows:

$$z = \underbrace{a^{m-1}}_u \underbrace{a}_v \underbrace{\varepsilon}_w \underbrace{b}_x \underbrace{b^{m-1}}_y$$

This division is good since $vx = ab \neq \varepsilon$ and $|vwx| = |ab| \leq 2$.

Subwords v and x can be pumped arbitrarily many times: for every $i \geq 0$

$$uv^iwx^iy = a^{m-1} a^i \varepsilon b^i b^{m-1} = a^{m+i-1} b^{m+i-1}$$

is in language L .

Let's analyze in detail why $L = \{a^m b^m \mid m \geq 0\}$ satisfies the pumping lemma.

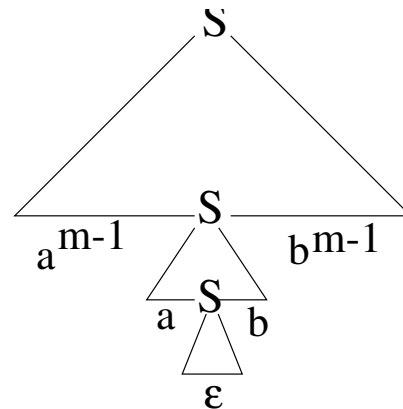
L is generated by a grammar with two productions

$$S \longrightarrow aSb \mid \varepsilon.$$

The word $a^m b^m$, $m \geq 1$, has a derivation

$$S \Longrightarrow^* a^{m-1} S b^{m-1} \Longrightarrow a^{m-1} a S b b^{m-1} \Longrightarrow a^{m-1} a \varepsilon b b^{m-1}$$

corresponding to the derivation tree



We used three subderivations

$$S \Longrightarrow^* a^{m-1} S b^{m-1},$$

$$S \Longrightarrow^* a S b,$$

$$S \Longrightarrow^* \varepsilon.$$

The middle subderivation $S \Longrightarrow^* a S b$ derives from variable S the variable itself.

We used three subderivations

$$S \Longrightarrow^* a^{m-1} S b^{m-1},$$

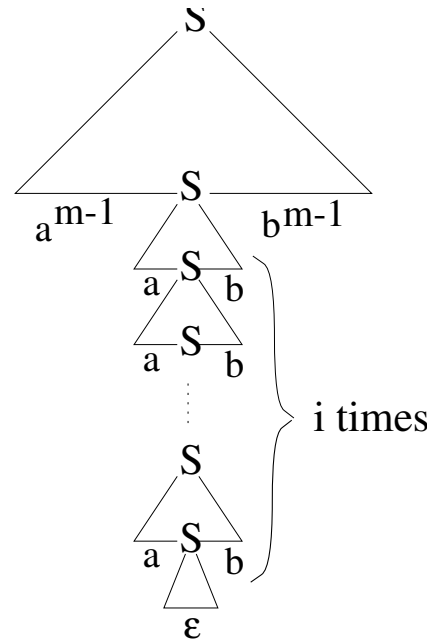
$$S \Longrightarrow^* aSb,$$

$$S \rightrightarrows^* \mathcal{E}.$$

The middle subderivation $S \Longrightarrow^* aSb$ derives from variable S the variable itself. It means that the subderivation can be iterated arbitrarily many times. Repeating it i times gives

$$S \Rightarrow^* a^{m-1} S b^{m-1} \Rightarrow^i a^{m-1} a^i S b^i b^{m-1} \Rightarrow a^{m-1} a^i \varepsilon b^i b^{m-1}$$

corresponding to the derivation tree



In other words, pumping i times words $v = a$ and $x = b$ corresponds to iterating the middle subderivation i times.

Let us prove that the argument above can be done for any context-free language.

Proof of the pumping lemma. Idea: The proof is based on the fact that during derivations of sufficiently long words, some variable derives a sentential form containing the variable itself. In other words, we have derivations

$$\begin{aligned} S &\Rightarrow^* uAy \\ A &\Rightarrow^* vAx \\ A &\Rightarrow^* w \end{aligned}$$

for some words u, v, w, x, y and variable A . But the derivation $A \Rightarrow^* vAx$ may be repeated arbitrarily many times, say i times:

$$\begin{aligned} S &\Rightarrow^* uAy \Rightarrow^* uvAxy \Rightarrow^* uvvAxxxy \Rightarrow^* \dots \\ &\dots \Rightarrow^* uv^i Ax^i y \Rightarrow^* uv^i wx^i y \end{aligned}$$

This shows that uv^iwx^iy is in language L for every $i \geq 0$.

Let us prove that the argument above can be done for any context-free language.

Proof of the pumping lemma. Idea: The proof is based on the fact that during derivations of sufficiently long words, some variable derives a sentential form containing the variable itself. In other words, we have derivations

$$\begin{aligned} S &\Rightarrow^* uAy \\ A &\Rightarrow^* vAx \\ A &\Rightarrow^* w \end{aligned}$$

for some words u, v, w, x, y and variable A . But the derivation $A \Rightarrow^* vAx$ may be repeated arbitrarily many times, say i times:

$$\begin{aligned} S &\Rightarrow^* uAy \Rightarrow^* uvAxy \Rightarrow^* uvvAxxxy \Rightarrow^* \dots \\ &\dots \Rightarrow^* uv^i Ax^i y \Rightarrow^* uv^i wx^i y \end{aligned}$$

This shows that uv^iwx^iy is in language L for every $i \geq 0$.

A precise proof: Let G be a grammar in Chomsky normal form such that $L = L(G)$ (or $L = L(G) \cup \{\varepsilon\}$ if $\varepsilon \in L$).

Define: the **depth** of a derivation tree = the length of the longest path from the root to a leaf.

Lemma. If a derivation tree by a CNF grammar for a word w has depth d then $|w| \leq 2^{d-1}$.

Proof.

Lemma. If a derivation tree by a CNF grammar for a word w has depth d then $|w| \leq 2^{d-1}$.

Proof. Mathematical induction on d :

1° (**base case**) If $d = 1$ then the tree must be the trivial one:

$$\begin{array}{c} A \\ | \\ a \end{array}$$

and the word it derives has length 2^{d-1} .

Lemma. If a derivation tree by a CNF grammar for a word w has depth d then $|w| \leq 2^{d-1}$.

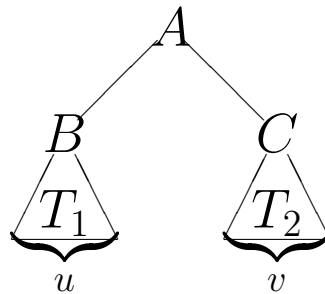
Proof. Mathematical induction on d :

1° **(base case)** If $d = 1$ then the tree must be the trivial one:



and the word it derives has length 2^{d-1} .

2° **(inductive step)** Let $d > 1$. The tree has the form



and $w = uv$.

Lemma. If a derivation tree by a CNF grammar for a word w has depth d then $|w| \leq 2^{d-1}$.

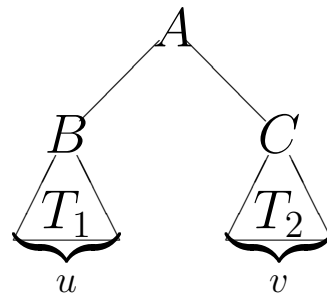
Proof. Mathematical induction on d :

1° (**base case**) If $d = 1$ then the tree must be the trivial one:



and the word it derives has length 2^{d-1} .

2° (**inductive step**) Let $d > 1$. The tree has the form



and $w = uv$. The depths of the trees T_1 and T_2 are at most $d - 1$, so that (by the inductive hypothesis) $|u| \leq 2^{d-2}$ and $|v| \leq 2^{d-2}$. Therefore

$$|w| = |u| + |v| \leq 2^{d-2} + 2^{d-2} = 2^{d-1}.$$

Let k be the **number of variables** in G , a CNF grammar for L .

Let us choose the number n in the pumping lemma as $n = 2^k$.

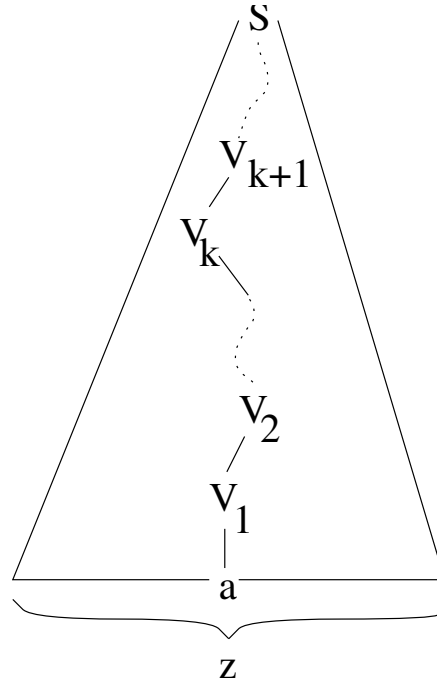
Let $z \in L$ be any word such that $|z| \geq n$, and let T be a derivation tree for z .

The depth of T has to be at least $k + 1$, since a tree of smaller depth can yield only shorter words.

Pick one maximum length path π from the root to a leaf in T . Reading from the leaf up, let

$$a, V_1, V_2, V_3, \dots, V_{k+1}$$

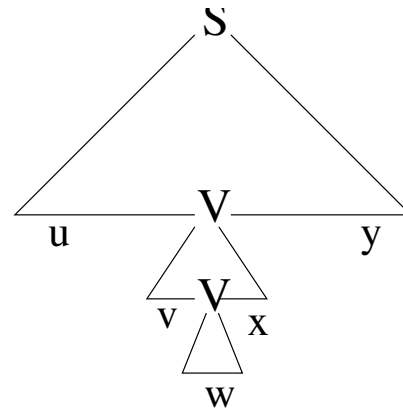
be the first $k + 2$ labels along path π :



Since the grammar has only k variables, it follows from the pigeon hole principal that two of the variables $V_1, V_2, \dots, V_{k+1}$ must be identical, say

$$V_s = V_t = V, \text{ for } s < t.$$

So the derivation tree for z looks like:

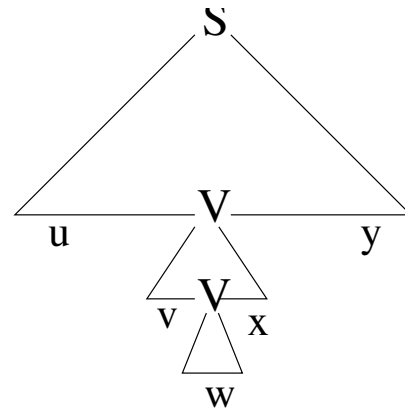


We have valid derivations

$$\begin{aligned}
 S &\Rightarrow^* uVy \\
 V &\Rightarrow^* vVx \\
 V &\Rightarrow^* w.
 \end{aligned}$$

Clearly $vx \neq \varepsilon$: Since the grammar G is in CNF, the lower V must have a sibling, and the yield of that sibling is part of either v or x .

So the derivation tree for z looks like:



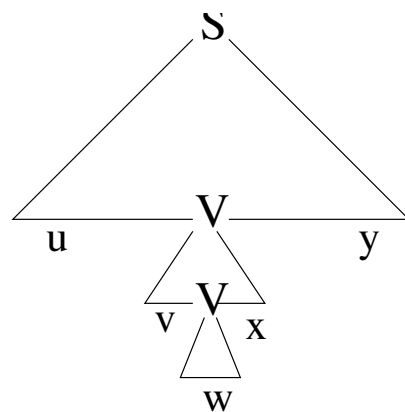
We have valid derivations

$$\begin{aligned}
 S &\Rightarrow^* uVy \\
 V &\Rightarrow^* vVx \\
 V &\Rightarrow^* w.
 \end{aligned}$$

Clearly $vx \neq \varepsilon$: Since the grammar G is in CNF, the lower V must have a sibling, and the yield of that sibling is part of either v or x .

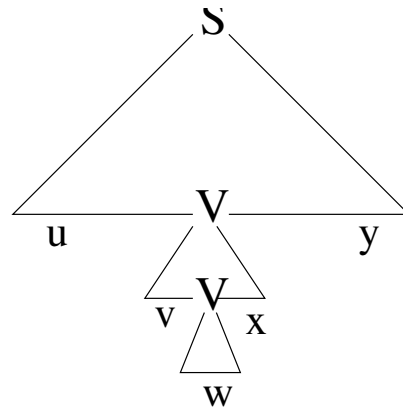
Now, there are derivations for words uv^iwx^iy for all $i \geq 0$:

$$\begin{aligned}
 S &\Rightarrow^* uVy \Rightarrow^* uvVxy \Rightarrow^* uv^2Vx^2y \Rightarrow^* \dots \\
 &\dots \Rightarrow^* uv^iVx^iy \Rightarrow^* uv^iwx^iy
 \end{aligned}$$



To complete the proof we still have to show that

$$|vwx| \leq n = 2^k.$$



To complete the proof we still have to show that

$$|vwx| \leq n = 2^k.$$

But vwx it is the yield of the subtree rooted at V_t (=the higher V), and the depth of that subtree is at most $k + 1$.

(If there would be longer path from V_t to a leaf, then that path combined with the path from the root S to V_t would be longer than the longest path π , a contradiction.)

We know how long can the yields of trees of depth $\leq k + 1$ be: they can have at most 2^k letters.

We use the pumping lemma to prove that certain languages are **not context-free**. It works the same way as the pumping lemma for regular languages. To show that a language is not a CFL we show that it does not satisfy the pumping lemma. So we are more interested in the negation of the pumping lemma.

Here is the mathematical formulation for the pumping lemma:

$$\begin{aligned} &(\exists n) \\ &(\forall z \in L, |z| \geq n) \\ &(\exists u, v, w, x, y : z = uvwxy, |vwx| \leq n, vx \neq \varepsilon) \\ &(\forall i \geq 0) \\ &uv^iwx^iy \in L, \end{aligned}$$

Its negation is the statement:

$$\begin{aligned} &(\forall n) \\ &(\exists z \in L, |z| \geq n) \\ &(\forall u, v, w, x, y : z = uvwxy, |vwx| \leq n, vx \neq \varepsilon) \\ &(\exists i \geq 0) \\ &uv^iwx^iy \notin L. \end{aligned}$$

Here is the mathematical formulation for the pumping lemma:

$$\begin{aligned} &(\exists n) \\ &(\forall z \in L, |z| \geq n) \\ &(\exists u, v, w, x, y : z = uvwxy, |vwx| \leq n, vx \neq \varepsilon) \\ &(\forall i \geq 0) \\ &uv^iwx^iy \in L, \end{aligned}$$

Its negation is the statement:

$$\begin{aligned} &(\forall n) \\ &(\exists z \in L, |z| \geq n) \\ &(\forall u, v, w, x, y : z = uvwxy, |vwx| \leq n, vx \neq \varepsilon) \\ &(\exists i \geq 0) \\ &uv^iwx^iy \notin L. \end{aligned}$$

To prove that a language is not context-free we do the following two steps:

- (1) For every n select a word $z \in L, |z| \geq n$,
- (2) for any division $z = uvwxy$ of z into five segments satisfying $|vwx| \leq n$ and $vx \neq \varepsilon$, find a number $i \geq 0$ such that

$$uv^iwx^iy$$

is **not** in the language L .

Example. Consider the language

$$L = \{a^m b^m c^m \mid m \geq 1\}.$$

Let us prove that L is **not** context-free.

Example. Consider the language

$$L = \{a^m b^m c^m \mid m \geq 1\}.$$

Let us prove that L is **not** context-free.

(1) For any given n choose word $z =$

Example. Consider the language

$$L = \{a^m b^m c^m \mid m \geq 1\}.$$

Let us prove that L is **not** context-free.

(1) For any given n choose word $z = a^n b^n c^n$. We have $z \in L$ and $|z| \geq n$, as required.

Example. Consider the language

$$L = \{a^m b^m c^m \mid m \geq 1\}.$$

Let us prove that L is **not** context-free.

(1) For any given n choose word $z = a^n b^n c^n$. We have $z \in L$ and $|z| \geq n$, as required.

(2) Let $z = uvwxy$ be an arbitrary division of z into 5 segments such that

$$|vwx| \leq n \quad \text{and} \quad vx \neq \varepsilon.$$

We must analyze possible choices of u, v, w, x, y : Since $|vwx| \leq n$, words v and x cannot contain both letters a and c . (In z , letters a are separated from all c 's by a string of n b 's.)

Example. Consider the language

$$L = \{a^m b^m c^m \mid m \geq 1\}.$$

Let us prove that L is **not** context-free.

(1) For any given n choose word $z = a^n b^n c^n$. We have $z \in L$ and $|z| \geq n$, as required.

(2) Let $z = uvwxy$ be an arbitrary division of z into 5 segments such that

$$|vwx| \leq n \quad \text{and} \quad vx \neq \varepsilon.$$

We must analyze possible choices of u, v, w, x, y : Since $|vwx| \leq n$, words v and x cannot contain both letters a and c . (In z , letters a are separated from all c 's by a string of n b 's.)

So letter a or c does not exist in v and x . Therefore uv^2wx^2y contains more some letters than others: By adding one v and x we have increased the number of some letters, while the number of one letter has remained n .

Conclusion: Choice $i = 2$ gives

$$uv^i wx^i y \notin L.$$

Another example.

$$L = \{a^m b^k c^m d^k \mid m, k \geq 1\}$$