Advanced Automata Theory 1
Chomsky Hierarchy and Grammars

Frank Stephan

Department of Computer Science

Department of Mathematics

National University of Singapore

fstephan@comp.nus.edu.sg

Languages

Language = Set of Strings over an Alphabet.
Alphabet Σ, for example $\Sigma = \{0, 1, 2\}$. Always finite.

Finite languages

$L_1 = \emptyset$, no elements.
$L_2 = \{\varepsilon\}$, set consisting of empty string.
$L_3 = \{00, 01, 02, 10, 11, 12, 20, 21, 22\}$, all elements of length 2.
$L_4 = \{\varepsilon, 0, 00, 000, 0000\}$, all strings of 0s up to length 4.
$L_5 = \{01, 001, 02, 002\}$, all strings consisting of one or two 0s followed by a 1 or 2.

Operations with Languages

Union:

$\mathbf{L} \cup \mathbf{H} = \{\mathbf{u} : \mathbf{u} \in \mathbf{L} \vee \mathbf{u} \in \mathbf{H}\}$;

$\{\mathbf{00}, \mathbf{01}, \mathbf{02}\} \cup \{\mathbf{01}, \mathbf{11}, \mathbf{21}\} = \{\mathbf{00}, \mathbf{01}, \mathbf{02}, \mathbf{11}, \mathbf{21}\}$;

$\{\mathbf{0}, \mathbf{00}, \mathbf{000}\} \cup \{\mathbf{00}, \mathbf{000}, \mathbf{0000}\} = \{\mathbf{0}, \mathbf{00}, \mathbf{000}, \mathbf{0000}\}$.

Intersection:

$\mathbf{L} \cap \mathbf{H} = \{\mathbf{u} : \mathbf{u} \in \mathbf{L} \wedge \mathbf{u} \in \mathbf{H}\}$;

$\{\mathbf{0}, \mathbf{00}, \mathbf{000}\} \cap \{\mathbf{00}, \mathbf{000}, \mathbf{0000}\} = \{\mathbf{00}, \mathbf{000}\}$;

$\{\mathbf{00}, \mathbf{01}, \mathbf{02}\} \cap \{\mathbf{01}, \mathbf{11}, \mathbf{21}\} = \{\mathbf{01}\}$.

Set Difference:

$\mathbf{L} - \mathbf{H} = \{\mathbf{u} : \mathbf{u} \in \mathbf{L} \wedge \mathbf{u} \notin \mathbf{H}\}$;

$\{\mathbf{00}, \mathbf{01}, \mathbf{02}\} - \{\mathbf{01}, \mathbf{11}, \mathbf{21}\} = \{\mathbf{00}, \mathbf{02}\}$.

Concatenation:

$\mathbf{000} \cdot \mathbf{1122} = \mathbf{0001122}$;

$\mathbf{L} \cdot \mathbf{H} = \{\mathbf{v} \cdot \mathbf{w} : \mathbf{v} \in \mathbf{L} \wedge \mathbf{w} \in \mathbf{H}\}$;

$\{\mathbf{0}, \mathbf{00}\} \cdot \{\mathbf{1}, \mathbf{2}\} = \{\mathbf{01}, \mathbf{001}, \mathbf{02}, \mathbf{002}\}$.

Kleene Star and Plus

Definition

$$\mathbf{L}^* = \{\varepsilon\} \cup \mathbf{L} \cup \mathbf{L} \cdot \mathbf{L} \cup \mathbf{L} \cdot \mathbf{L} \cdot \mathbf{L} \cup \ldots$$
$$= \{\mathbf{w_1} \cdot \mathbf{w_2} \cdot \ldots \cdot \mathbf{w_n} : \mathbf{n} \geq \mathbf{0} \wedge \mathbf{w_1}, \mathbf{w_2}, \ldots, \mathbf{w_n} \in \mathbf{L}\};$$
$$\mathbf{L}^+ = \mathbf{L} \cup \mathbf{L} \cdot \mathbf{L} \cup \mathbf{L} \cdot \mathbf{L} \cdot \mathbf{L} \cup \ldots$$
$$= \{\mathbf{w_1} \cdot \mathbf{w_2} \cdot \ldots \cdot \mathbf{w_n} : \mathbf{n} > \mathbf{0} \wedge \mathbf{w_1}, \mathbf{w_2}, \ldots, \mathbf{w_n} \in \mathbf{L}\}.$$

Examples

$\emptyset^* = \{\varepsilon\}$.

$\mathbf{\Sigma}^*$ is the set of all words over $\mathbf{\Sigma}$.

$\{\mathbf{0}\}^* = \{\varepsilon, \mathbf{0}, \mathbf{00}, \mathbf{000}, \mathbf{0000}, \ldots\}$.

$\{\mathbf{00}, \mathbf{01}, \mathbf{10}, \mathbf{11}\}^*$ are all binary words of even length.

$\varepsilon \in \mathbf{L}^+$ iff $\varepsilon \in \mathbf{L}$.

Notation

Often $\mathbf{a}^*$ in place of $\{\mathbf{a}\}^*$ and $\mathbf{abc}^*$ in place of $\{\mathbf{ab}\} \cdot \{\mathbf{c}\}^*$;
For single variable $\mathbf{w}$, $\mathbf{w}^*$ is $\{\mathbf{w}\}^*$ and $\mathbf{w} \cdot \mathbf{L}$ is $\{\mathbf{w}\} \cdot \mathbf{L}$.

Regular Languages

Regular expressions are either finite sets listed by their elements or obtained from other regular expressions by forming the Kleene star, Kleene plus, union, intersection, set-difference or concatenation.

A language is regular iff it can be described by a regular expression.

Regular sets have many different regular expressions.

For example, $\{0, 00\} \cdot \{1, 2\}$ and $\{01, 001, 02, 002\}$ describe the same set. Also 0^* and $(00)^* \cup 0 \cdot (00)^*$ describe the same set.

Intersections and set difference are traditionally not used in regular expressions, as they can be replaced by combining other operations.

The complement of a language L is $\Sigma^* - L$.

Quiz

Which of the following regular expressions describe the same set?

1. $\{00, 000\}^+$;
2. $\{000, 0000\}^+$;
3. $00 \cdot 0^*$;
4. $000 \cdot 0^*$;
5. $\{000, 0000\} \cup (000000 \cdot 0^*)$;
6. $\{00, 01, 02, 10, 11, 12\}$;
7. $0^*1^*2^*$;
8. $(0^*1^*2^*)^*$;
9. $(\{0, 1\} \cdot \{0, 1, 2\}^*) \cap (\{0, 1, 2\} \cdot \{0, 1, 2\})$.

Grammars

Grammar $(\mathbf{N}, \mathbf{\Sigma}, \mathbf{P}, \mathbf{S})$ describes how to generate the words in a language; the language $\mathbf{L}$ of a grammar consists of all the words in $\mathbf{\Sigma}^*$ which can be generated.

$\mathbf{N}$: Non-terminal alphabet, disjoint to $\mathbf{\Sigma}$.

$\mathbf{S} \in \mathbf{N}$ is the start symbol.

$\mathbf{P}$ consists of rules $\mathbf{l} \to \mathbf{r}$ with each rule having at least one symbol of $\mathbf{N}$ in the word $\mathbf{l}$.

$\mathbf{v} \Rightarrow \mathbf{w}$ iff there are $\mathbf{x}, \mathbf{y}$ and rule $\mathbf{l} \to \mathbf{r}$ in $\mathbf{P}$ with $\mathbf{v} = \mathbf{xly}$ and $\mathbf{w} = \mathbf{xry}$. $\mathbf{v} \Rightarrow^* \mathbf{w}$: none, one or many such steps.

The grammar with $\mathbf{N} = \{\mathbf{S}\}$, $\mathbf{\Sigma} = \{\mathbf{0}, \mathbf{1}\}$ and $\mathbf{P} = \{\mathbf{S} \to \mathbf{SS}, \mathbf{S} \to \mathbf{0}, \mathbf{S} \to \mathbf{1}\}$ permits to generate all nonempty binary strings.

$\mathbf{S} \Rightarrow \mathbf{SS} \Rightarrow \mathbf{SSS} \Rightarrow \mathbf{0SS} \Rightarrow \mathbf{01S} \Rightarrow \mathbf{011}$.

Examples

Example 1.7

At least three symbols, 0s followed by 1s, at least one 0 and one 1.

$\mathbf{N} = \{\mathbf{S}, \mathbf{T}\}$, $\mathbf{\Sigma} = \{\mathbf{0}, \mathbf{1}\}$, startsymbol $\mathbf{S}$, $\mathbf{P}$ has $\mathbf{S} \rightarrow \mathbf{0T1}$, $\mathbf{T} \rightarrow \mathbf{0T}$, $\mathbf{T} \rightarrow \mathbf{T1}$, $\mathbf{T} \rightarrow \mathbf{0}$, $\mathbf{T} \rightarrow \mathbf{1}$.

Example 1.8

All words with as many 0s as 1s.

$\mathbf{N} = \{\mathbf{S}\}$, $\mathbf{\Sigma} = \{\mathbf{0}, \mathbf{1}\}$, $\mathbf{S} \rightarrow \mathbf{SS}|\mathbf{0S1}|\mathbf{1S0}|\varepsilon$.

The symbol $|$ separates alternatives.

Example 1.9

All words of odd length.

$\mathbf{N} = \{\mathbf{S}, \mathbf{T}\}$, $\mathbf{\Sigma} = \{\mathbf{0}, \mathbf{1}, \mathbf{2}\}$, startsymbol $\mathbf{S}$, $\mathbf{S} \rightarrow \mathbf{0T}|\mathbf{1T}|\mathbf{2T}|\mathbf{0}|\mathbf{1}|\mathbf{2}$, $\mathbf{T} \rightarrow \mathbf{0S}|\mathbf{1S}|\mathbf{2S}$.

The Chomsky Hierarchy

Grammar $(\mathbf{N}, \mathbf{\Sigma}, \mathbf{P}, \mathbf{S})$ generating $\mathbf{L}$.

CH0: No restriction. Generates all recursively enumerable languages.

CH1 (context-sensitive): Every rule is of the form $\mathbf{uAw} \to \mathbf{uvw}$ with $\mathbf{A} \in \mathbf{N}$, $\mathbf{u}, \mathbf{v}, \mathbf{w} \in (\mathbf{N} \cup \mathbf{\Sigma})^*$ and $\mathbf{v} = \varepsilon$ is only possible if $\mathbf{A} = \mathbf{S}$ and $\mathbf{S}$ does not occur on any right side of a rule.

Easier formalisation: If $\mathbf{l} \to \mathbf{r}$ is a rule then $|\mathbf{l}| \leq |\mathbf{r}|$, that is, $\mathbf{r}$ is at least as long as $\mathbf{l}$. Special rule (as above) for the case that $\varepsilon \in \mathbf{L}$.

CH2 (context-free): Every rule is of the form $\mathbf{A} \to \mathbf{w}$ with $\mathbf{A} \in \mathbf{N}$ and $\mathbf{w} \in (\mathbf{N} \cup \mathbf{\Sigma})^*$.

CH3 (regular): Every rule is of the form $\mathbf{A} \to \mathbf{wB}$ or $\mathbf{A} \to \mathbf{w}$ with $\mathbf{A}, \mathbf{B} \in \mathbf{N}$ and $\mathbf{w} \in \mathbf{\Sigma}^*$.

Examples

Regular grammar for Example 1.7:
$\mathbf{N = \{S, T\}}$, $\mathbf{\Sigma = \{0, 1\}}$, startsymbol $\mathbf{S}$, $\mathbf{S \to 0S|00T|01T}$,
$\mathbf{T \to 1T|1}$.

Grammar for Example 1.8 is context-free.

Grammar for Example 1.9 is regular.

Example 1.13.
Context-Sensitive Grammar for $\mathbf{\{0^n 1^n 2^n : n \in \mathbb{N}\}}$.
$\mathbf{N = \{S, T, U\}}$, $\mathbf{\Sigma = \{0, 1, 2\}}$, startsymbol $\mathbf{S}$,
$\mathbf{S \to 012|0T12|\varepsilon}$, $\mathbf{T \to 0T1U|01U}$, $\mathbf{U1 \to 1U}$, $\mathbf{U2 \to 22}$.

Regular Grammar $\Rightarrow$ Expression

Regular grammar $(\{S, T\}, \{0, 1, 2, 3\}, P, S)$ with $S \to 0S|1T|2$ and $T \to 0T|1S|3$.
Let $L_S = \{w : (S \to w) \in P\} = \{2\}$ and $L_T = \{3\}$.
Let $L_{S,S} = \{w : (S \to wS) \in P\} = \{0\}$, $L_{S,T} = \{1\}$, $L_{T,S} = \{1\}$, $L_{T,T} = \{0\}$.

Regular Expression:
$(L_{S,S})^* \cdot (L_{S,T} \cdot (L_{T,T})^* \cdot L_{T,S} \cdot (L_{S,S})^*)^* \cdot (L_S \cup L_{S,T} \cdot (L_{T,T})^* \cdot L_T)$
giving $0^* \cdot (10^*10^*)^* \cdot (2 \cup 10^*3)$.

Equivalent expression:
$(L_{S,S} \cup L_{S,T} \cdot (L_{T,T})^* \cdot L_{T,S})^* \cdot (L_S \cup L_{S,T} \cdot (L_{T,T})^* \cdot L_T)$
giving $(0 \cup 10^*1)^* \cdot (2 \cup 10^*3)$.

Regular Expression $\Rightarrow$ Grammar

Given $(\{0,1\}^* \cdot 2 \cdot \{0,1\}^* \cdot 2) \cup \{0,2\}^* \cup \{1,2\}^*$.

Choose Non-Terminals S, T, U, V, W with
$L_S = L_T \cup L_V \cup L_W$;
$L_T = \{0,1\}^* \cdot 2 \cdot \{0,1\}^* \cdot 2 = \{0,1\}^* \cdot 2 \cdot L_U$;
$L_U = \{0,1\}^* \cdot 2$;
$L_V = \{0,2\}^*$;
$L_W = \{1,2\}^*$.

Grammar $(\{S,T,U,V,W\}, \{0,1,2\}, P, S)$ with these rules:
$S \to T|V|W$,
$T \to 0T|1T|2U$,
$U \to 0U|1U|2$,
$V \to 0V|2V|\varepsilon$,
$W \to 1W|2W|\varepsilon$.

Quiz 1.10 and 1.16

Quiz 1.10: Make a regular grammar for $\mathbf{0^*10^*10^*10^*10^*20^*}$.

Quiz 1.16: Consider the language $\mathbf{L = \{00, 11, 22\} \cdot \{33\}^*}$:
(a) Make a regular grammar for $\mathbf{L}$;
(b) Make a regular grammar for $\mathbf{H = L^*}$.

The Pumping Lemma

Theorem 1.19 (a)
Let $\mathbf{L}$ be a regular language. There is a constant $\mathbf{k}$ such that every $\mathbf{w} \in \mathbf{L}$ with $|\mathbf{w}| > \mathbf{k}$ equals to $\mathbf{xyz}$ with $\mathbf{y} \neq \varepsilon$ and $|\mathbf{xy}| \leq \mathbf{k}$ and $\mathbf{xy^*z} \subseteq \mathbf{L}$.

Tighter versions will be shown later.

Corollary 1.20.
If $\mathbf{L}$ is an infinite regular language then almost all words of $\mathbf{L}$ can be brought into the form $\mathbf{xyz}$ with $\mathbf{y} \neq \varepsilon$ and $\mathbf{xy^*z} \subseteq \mathbf{L}$.

Example
$\mathbf{L = 0110 \cdot \{2, 3\}^* \cup 001100 \cdot \{22, 33\}^* \cdot 11 \cup 0011001100 \cdot \{2, 3\}}$.
Then constant $\mathbf{k}$ is 11.
If $\mathbf{w} \in \mathbf{L}$ and $|\mathbf{w}| > \mathbf{11}$ then there are at least two occurrences of $\mathbf{2, 3}$ in $\mathbf{w}$.
So split $\mathbf{w}$ into $\mathbf{xyz}$ such that $\mathbf{y}$ is the first block of two digits from $\mathbf{2, 3}$ occurring in $\mathbf{w}$. Then $\mathbf{xy^*z} \subseteq \mathbf{L}$.

Pumping Lemma 1.19 (a) as a Game

Player Anke wants to show pumping condition is true; Player Boris wants to show it is false.

1. Anke selects a pumping constant k.
2. Boris selects a word $z \in L$ with $|z| \geq k$; if such a word does not exist, Anke has won.
3. Anke splits word z into three parts u, v, w with $z = uvw$ such that $|v| \leq k$ and $|v| \geq 1$.
4. Boris selects $h \in \mathbb{N}$; note that $h = 0$ is possible.
5. If $uv^h w \in L$ then Anke wins else Boris wins.

The language L satisfies the pumping condition ("L satisfies the pumping lemma") iff Anke can play for the given L the game such that she always wins, that is, iff Anke has a winning strategy for the above game.

The Pumping Lemma says that whenever L is regular then Anke has a winning strategy for the game above.

Structural Induction

To show that all regular sets satisfy the Pumping Lemma, one does the following.

Show that $\mathbf{H}$ satisfies the Pumping Lemma for all finite $\mathbf{H}$.

Show that if $\mathbf{H}$ satisfies the Pumping Lemma, so does $\mathbf{H}^*$.

Show that if $\mathbf{H_1}, \mathbf{H_2}$ satisfy the Pumping Lemma, so does $\mathbf{H_1} \cdot \mathbf{H_2}$.

Show that if $\mathbf{H_1}, \mathbf{H_2}$ satisfy the Pumping Lemma, so does $\mathbf{H_1} \cup \mathbf{H_2}$.

Other operations (intersection, set difference) can be ignored, as one can make all regular languages without using them.

How to Prove these Items

Finite Sets: Choose constant larger than longest word; so no word applies to be pumped.

Kleene Star: Pumping Constant $\mathbf{c}$ remains the same. Given $\mathbf{w_1 w_2 \ldots w_n \in H^*}$, either $\mathbf{w_1}$ is shorter than $\mathbf{c}$ and can be repeated or $\mathbf{w_1}$ is longer than $\mathbf{c}$ and one can pump inside $\mathbf{w_1}$.

Concatenation: If $\mathbf{H_1, H_2}$ have pumping constants $\mathbf{c_1, c_2}$, then consider $\mathbf{v \cdot w}$ longer than $\mathbf{c_1 + c_2}$. Either $\mathbf{v}$ is longer than $\mathbf{c_1}$ or $\mathbf{w}$ is longer than $\mathbf{c_2}$. That one of $\mathbf{v, w}$ which overshoots the length can be pumped.

Union: The pumping constant of the union is the maximum of the given constants. Any word $\mathbf{u \in H_1 \cup H_2}$ longer than this maximum can be pumped, as the pumped versions $\mathbf{xy^*z}$ is a subset of either $\mathbf{H_1}$ or $\mathbf{H_2}$, respectively, and therefore also a subset of the union.

Pumping Position and Length

Examples 1.25

Let $\mathbf{L} = \{\mathbf{w} \in \{\mathbf{0},\mathbf{1}\}^* : \mathbf{w}$ has as many $\mathbf{0}$s as $\mathbf{1}$s$\}$.

Satisfies Pumping-Lemma Without Constraint on Pumping-Position and Length (Corollary 1.20).

Given $\mathbf{w} \in \{\mathbf{0},\mathbf{1}\}^* - \{\mathbf{0}\}^* - \{\mathbf{1}\}^*$. Then $\mathbf{w} = \mathbf{xyz}$ with $\mathbf{y} \in \{\mathbf{01},\mathbf{10}\}$.

If $\mathbf{w} \in \mathbf{L}$ then $\mathbf{xy^*z} \subseteq \mathbf{L}$.

Does not Satisfy Pumping Lemma with Constraint on Pumping-Position and Length (Theorem 1.19 (a)).

Let $\mathbf{k}$ be the pumping constant and consider $\mathbf{0^{k+1}1^{k+1}}$. Pumping before position $\mathbf{k}$ expands or reduces the number of $\mathbf{0}$s without adjusting the number of $\mathbf{1}$s the same way.

Context-Free Languages

Pumping-Lemma for Context-Free Languages (Thm 1.19 (b))

Assume that $\mathbf{L}$ is a context-free language. Then there is a constant $\mathbf{k}$ such that for all $\mathbf{u} \in \mathbf{L}$ with $|\mathbf{u}| > \mathbf{k}$ there is a representation $\mathbf{vwxyz}$ of $\mathbf{u}$ with $|\mathbf{wxy}| \leq \mathbf{k}$ and $\mathbf{w} \neq \varepsilon \vee \mathbf{y} \neq \varepsilon$ and $\mathbf{vw^n xy^n z} \in \mathbf{L}$ for all $\mathbf{n} \in \mathbb{N}$.

Applications

Showing that certain languages are not context-free or regular.

$\mathbf{L} = \{\mathbf{u} : \mathbf{u}$ is a decimal number where every digit appears as often as the other digits$\}$.
This language is not context-free.

$\mathbf{L} = \{\mathbf{3^n 7^n} : \mathbf{n} \in \{\mathbf{1}, \mathbf{2}, \mathbf{3}, \ldots\}\}$.
This language is context-free but not regular.

Pumping Lemma 1.19 (b) as a Game

Player Anke wants to show pumping condition is true;
Player Boris wants to show it is false.

1. Anke selects a pumping constant k.
2. Boris selects a word $u \in L$ with $|u| \geq k$; if such a word does not exist, Anke has won.
3. Anke splits word z into five parts v, w, x, y, z with $u = vwxyz$ such that $|wxy| \leq k$ and $|w| \geq 1$.
4. Boris selects $h \in \mathbb{N}$; note that $h = 0$ is possible.
5. If $vw^h xy^h z \in L$ then Anke wins else Boris wins.

The language L satisfies the pumping condition ("L satisfies the pumping lemma") iff Anke can play for the given L the game such that she always wins, that is, iff Anke has a winning strategy for the above game.

The Pumping Lemma says that whenever L is context-free then Anke has a winning strategy for the game above.

Primes

Example 1.24

The set $\mathbf{L} = \{\mathbf{0^p} : \mathbf{p}$ is a prime$\}$ is not context-free.

Let $\mathbf{k}$ be the pumping constant and $\mathbf{p}$ be a prime number larger than $\mathbf{k}$.

Now $\mathbf{0^p} = \mathbf{vwxyz}$ with $\mathbf{wy} \neq \varepsilon$ and $\mathbf{vw^rxy^rz} \in \mathbf{L}$ for all $\mathbf{r}$.

Let $\mathbf{q} = |\mathbf{wy}|$, note that $\mathbf{q} > \mathbf{0}$.

Now $\mathbf{vw^{p+1}xy^{p+1}z} \in \mathbf{L}$ and has length $\mathbf{p} + \mathbf{p} * \mathbf{q}$.

This is $\mathbf{p} * (\mathbf{1} + \mathbf{q})$ and is not a prime.

Hence $\mathbf{0^{p+p*q}} \notin \mathbf{L}$, a contradiction to the Pumping Lemma.

So $\mathbf{L}$ does not satisfy the Pumping Lemma for context-free languages.

Theorem 1.26

Let $\mathbf{L} \subseteq \{\mathbf{0}\}^*$.
The following conditions are equivalent for $\mathbf{L}$.

1. $\mathbf{L}$ is regular;
2. $\mathbf{L}$ is context-free;
3. $\mathbf{L}$ satisfies the Pumping Lemma for regular languages;
4. $\mathbf{L}$ satisfies the Pumping Lemma for context-free languages.

However, the bound on the length of the pumped word is necessary. The set $\{\mathbf{0^p} : \mathbf{p}$ is not a power of two$\}$ satisfies the Pumping Lemma without that bound.

Proof of Theorem 1.26

One shows the following: If a language $\mathbf{L} \subseteq \{0\}^*$ satisfies the context-free pumping lemma then it is regular.

Let $\mathbf{k_1, k_2, \ldots, k_\ell}$ be the possible lengthes of $\mathbf{wy}$ when pumping $\mathbf{vwxyz}$ as $\mathbf{vw^hxy^hz}$ for all $\mathbf{h} \in \mathbb{N}$. As there is only one symbol, one can write these words as $\mathbf{vxz(wy)^h}$. Now one can unify to a single pumping length $\mathbf{k = k_1 \cdot k_2 \cdot \ldots \cdot k_\ell}$. Let $\mathbf{F}$ be the finite set of $\mathbf{u}$ with $\mathbf{u} \cdot \{\mathbf{0^k}\}^* \not\subseteq \mathbf{L}$.

Now $\mathbf{L}$ satisfies for almost all $\mathbf{h} \in \mathbb{N}$, if $\mathbf{0^h} \in \mathbf{L}$ then $\mathbf{0^{h+k}} \in \mathbf{L}$. Let $\mathbf{G} = \{\mathbf{u} \in \{0\}^* : \mathbf{u} \cdot \{\mathbf{0^k}\}^* \subseteq \mathbf{L}$ and $\forall \mathbf{v} \in \mathbf{L}\,[\mathbf{u} \neq \mathbf{v} \cdot \mathbf{0^k}]\}$. Note that $\mathbf{G}$ contains for each remainder $\ell < \mathbf{k}$ at most one $\mathbf{u}$ with $|\mathbf{u}|\%\mathbf{k} = \ell$.

So $\mathbf{l}$ has a regular expression of the form $\mathbf{F} \cup \mathbf{G} \cdot \{\mathbf{0^h}\}^*$, where $\mathbf{G}$ is the finite set from above and $\mathbf{F}$ is the finite set of $\mathbf{u} \in \mathbf{L}$ with $\mathbf{u} \cdot \{\mathbf{0^k}\}^* \not\subseteq \mathbf{L}$. Thus $\mathbf{L}$ is regular.

Other directions

Clearly a regular set is context-free and furthermore satisfies the regular pumping lemma; a set which is either context-free or satisfies the regular pumping lemma, then also satisfies the context-free pumping lemma.

Now consider $\mathbf{L} = \{\mathbf{0^p} : \mathbf{p} > \mathbf{0}$ and $\mathbf{p}$ is not a power of $\mathbf{2}\}$. Assume now that $\mathbf{0^p}$ is given with $\mathbf{p}$ neither a power of $\mathbf{2}$ nor $\mathbf{p} \leq \mathbf{3}$. If $\mathbf{p}$ is a multiple of $\mathbf{3}$ and $\mathbf{p} \geq \mathbf{6}$, then $\mathbf{0^{p-3}} \cdot \{\mathbf{000}\}^*$ is a regular subset of $\mathbf{L}$. If $\mathbf{p} = \mathbf{2^i} + \mathbf{j}$ with $\mathbf{0} < \mathbf{j} < \mathbf{2^i}$ and $\mathbf{j} \neq \mathbf{2^{i-1}}$ then $\mathbf{p} - \mathbf{2^{i-1}}$ is neither a power of $\mathbf{2}$ nor any number of the form $\mathbf{p} + (\mathbf{h} - \mathbf{1}) \cdot \mathbf{2^{i-1}}$ is a power of $\mathbf{2}$ where $\mathbf{h} \in \mathbb{N}$ is arbitrary. Thus one can choose the pump as of length $\mathbf{2^{i-1}}$. Thus for all $\mathbf{0^p} \in \mathbf{L}$ except the shortest one $\mathbf{0^3}$, pumping up and down is both possible. So the pumping lemma without a bound on the length of the pump is satisfied.

Exercise 1.27

Exercise

Let $\mathbf{L} = \{\mathbf{0^n 1^n 2^n} : \mathbf{n} \in \mathbb{N}\}$,
$\mathbf{H} = \{\mathbf{0^n 1^m} : \mathbf{n^2} \leq \mathbf{m} \leq \mathbf{2n^2}\}$ and
$\mathbf{K} = \{\mathbf{0^n 1^m 2^k} : \mathbf{n} \cdot \mathbf{m} = \mathbf{k}\}$.

Show that these languages are not context-free using the Pumping Lemma for context-free languages.

Comment

For $\mathbf{L}$ this is a classical result and standard exercise in the field. This example often comes up and it is useful to remember it. It will be used in varied form for various further results. This exercise is mainly for students new to theory.

Exercise 1.28-1.30

Construct grammars for the following languages over the alphabet $\{0, 1\}$:

Exercise 1.28: A context-sensitive grammar for $\{10^n1 : n$ is a power of three$\}$.

Exercise 1.29: A context-sensitive grammar for $\{10^n1 : n$ is a non-trivial product$\}$.

Exercise 1.30: A context-free language for $\{uvw : |u| = |v| = |w|$ and $u \neq w\}$.

Exercises 1.31-1.33

Let $\mathbf{F(L)}$ be the set of all permutations of words in $\mathbf{L}$, so $\mathbf{F(\{0,00,011\}) = \{0,00,011,101,110\}}$.

Exercise 1.31: For which levels of the Chomsky hierarchy is there a regular $\mathbf{L}$ such that $\mathbf{F(L)}$ takes exactly the given level? Provide the languages and explain why they are on that level.

Exercise 1.32: Consider the following weaker version of the Pumping Lemma: Almost all words $\mathbf{u} \in \mathbf{L}$ can be split into $\mathbf{u} = \mathbf{vwxyz}$ such that $\mathbf{wy} \neq \varepsilon$ and $\forall \mathbf{n} \in \mathbb{N}\,[\mathbf{vw^n xy^n z} \in \mathbf{L}]$. Provide a regular language $\mathbf{L}$ such that $\mathbf{F(L)}$ satisfies this weaker version of the pumping Lemma but neither the context-free Pumping Lemma nor Corollary 1.20.

Exercise 1.33: Let $\mathbf{L} = \{\mathbf{0^n 1^m 2^k} : \mathbf{n} \neq \mathbf{m} \vee \mathbf{n} \neq \mathbf{k} \vee \mathbf{m} \neq \mathbf{k}\}$. Show that $\mathbf{L}$ satisfies Corollary 1.20 with pump length $\mathbf{1}$? If $\mathbf{H}$ satisfies Corollary 1.20 with pump length $\mathbf{1}$, does so $\mathbf{F(H)}$?

Exercises 1.34-1.36

Exercise 1.34: Let $\mathbf{G(L) = \{vw : wv \in L}$ and $\mathbf{v, w \in \Sigma^*\}}$. Provide all levels of the Chomsky hierarchy for which there is an $\mathbf{L}$ such that $\mathbf{G(L)}$ is regular.

Exercise 1.35: Let $\mathbf{L = \{w \in \{0,1,2,3\}^* :}$ if $\mathbf{a < b}$ then $\mathbf{b}$ occurs more frequently than $\mathbf{a\}}$. What is the exact level of $\mathbf{L}$ in the Chomsky hierarchy? Use grammars and pumping lemmas to prove the result.

Exercise 1.36: Let $\mathbf{L}$ be given by the grammar $\mathbf{(\{S\}, \{0,1\},}$ $\mathbf{\{S \to 01S|01, S0 \to 0S, S1 \to 1S, 0S \to S0, 1S \to S1\}, S)}$ Determine the level of $\mathbf{L}$ in the Chomsky hierarchy and determine all words up to length $\mathbf{6}$ in $\mathbf{L}$. Explain which words $\mathbf{L}$ contains.

Exercises 1.37-1.39

Exercise 1.37: Construct context-free grammars for the sets $\mathbf{L} = \{\mathbf{0^n 1^m 2^k} : \mathbf{n} < \mathbf{m} \vee \mathbf{m} < \mathbf{k}\}$, $\mathbf{H} = \{\mathbf{0^n 1^m 2^{n+m}} : \mathbf{n}, \mathbf{m} \in \mathbb{N}\}$ and $\mathbf{K} = \{\mathbf{w} \in \{\mathbf{0}, \mathbf{1}, \mathbf{2}\}^* : \mathbf{w}$ has a subword of the form $\mathbf{20^n 1^n 2}$ for some $\mathbf{n} > \mathbf{0}$ or $\mathbf{w} = \varepsilon\}$. Which of the versions of the Pumping Lemma (Theorem 1.19 (a), 1.19 (b), Corollary 1.20) do $\mathbf{L}, \mathbf{H}, \mathbf{K}$ satisfy?

Exercise 1.38: Let $\mathbf{L} = \{\mathbf{0^h 1^i 2^j 3^k} : (\mathbf{h} \neq \mathbf{i}$ and $\mathbf{j} \neq \mathbf{k})$ or $(\mathbf{h} \neq \mathbf{k}$ and $\mathbf{i} \neq \mathbf{j})\}$ be given. Construct a context-free grammar for $\mathbf{L}$ and determine which of versions of the Pumping Lemma $\mathbf{L}$ satisfies.

Exercise 1.39: Consider the linear grammar $(\{\mathbf{S}\}, \{\mathbf{0}, \mathbf{1}, \mathbf{2}, \mathbf{3}\}, \{\mathbf{S} \to \mathbf{00S} | \mathbf{S1} | \mathbf{S2} | \mathbf{3}\}, \mathbf{S})$ and construct for the language $\mathbf{L}$ generated by the grammar the following: a regular grammar for $\mathbf{L}$ and a regular expression for $\mathbf{L}$.

Grammars and Growth

For the following exercises, let $\mathbf{f(n)}$ be the number of words $\mathbf{w \in L}$ with $\mathbf{|w| < n}$. To answer the questions, either construct a grammar witnessing that such an $\mathbf{L}$ exists or prove that it cannot exist.

Exercise 1.40
Is there a context-free language $\mathbf{L}$ with $\mathbf{f(n) = \lfloor \sqrt{n} \rfloor}$?

Exercise 1.41
Is there a regular $\mathbf{L}$ with $\mathbf{f(n) = n(n+1)/2}$?

Exercise 1.42
Is there a context-sensitive $\mathbf{L}$ with $\mathbf{f(n) = n^n}$, where $\mathbf{0^0 = 0}$?

Exercise 1.43
Is there a regular $\mathbf{L}$ with $\mathbf{f(n) = (3^n - 1)/2 + \lfloor n/2 \rfloor}$?

Exercise 1.44
Is there a regular $\mathbf{L}$ with $\mathbf{f(n) = \lfloor n/3 \rfloor + \lfloor n/2 \rfloor}$?