On Lyndon words and Lyndon trees

Francesco Dolce

Graphs, Games, Optimization, Algorithms and TCS (Online) Seminar

3. května 2021

What to expect from this talk

1. **Lyndon words**
2. **Lyndon factorization**
3. **Infinite words**
4. **Generalized Lyndon words**
5. **Lyndon trees**

A few words about words

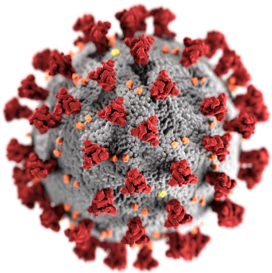
- $\mathit{str\check{c}}, \mathit{prst}, \mathit{skrz}, \mathit{krk} \in \{a, b, c, \ldots, \check{z}\}^*$

A few words about words

- *strč*, *prst*, *skrz*, *krk* $\in \{a, b, c, \dots, ž\}^*$
- $001, 101000, 010101010 \in \{0, 1\}^*$

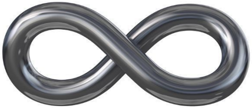
A few words about words

- *strč*, *prst*, *skrz*, *krk* $\in \{a, b, c, \ldots, ž\}^*$
- $001, 101000, 010101010 \in \{0,1\}^*$
- $\mathtt{ACGATACGGACATTACATATACG} \in \{\mathtt{A}, \mathtt{C}, \mathtt{G}, \mathtt{T}\}^+$



A few words about words

- $\mathit{str\check{c}}, \mathit{prst}, \mathit{skrz}, \mathit{krk} \in \{a, b, c, \ldots, \check{z}\}^*$
- $\texttt{001}, \texttt{101000}, \texttt{010101010} \in \{\texttt{0}, \texttt{1}\}^*$
- $\texttt{ACGATACGGACATTACATATACG} \in \{\texttt{A}, \texttt{C}, \texttt{G}, \texttt{T}\}^+$
- $\texttt{babaabaabaaabaaab} \cdots \in \{\texttt{a}, \texttt{b}\}^\omega$



A few words about words

- $\mathit{str\check{c}}, \mathit{prst}, \mathit{skrz}, \mathit{krk} \in \{a, b, c, \ldots, \check{z}\}^*$
- $001, 101000, 010101010 \in \{0, 1\}^*$
- $\mathtt{ACGATACGGACATTACATATACG} \in \{\mathtt{A}, \mathtt{C}, \mathtt{G}, \mathtt{T}\}^+$
- $\mathtt{babaabaabaaabaaab} \cdots \in \{\mathtt{a}, \mathtt{b}\}^\omega$

$$w = pfs$$

prefix   factor   suffix

The *length* $|w|$ of a finite word w is the total number of letters of w.

$\mathcal{A}^*$ = finite words, $\mathcal{A}^+$ = non-empty finite words, $\mathcal{A}^\omega$ = infinite words.

Oredered alphabets

Definition

Let us consider a total order $<$ on $\mathcal{A}$.
This order can be extended to $\mathcal{A}^*$, and it is called *lexicographical order*, by setting

$$u < v \iff \begin{cases} v = us & s \in \mathcal{A}^+ \\ \text{or} & \\ u = pas,\ v = pbt & p, s, t \in \mathcal{A}^*,\quad a, b \in \mathcal{A},\quad a < b \end{cases}$$

Example

If $\mathcal{A} = \{\mathtt{a}, \mathtt{b}, \mathtt{c}\}$ and $\mathtt{a} < \mathtt{b} < \mathtt{c}$, then

$$\mathtt{a} < \mathtt{aab} < \mathtt{ab} < \mathtt{aba} < \mathtt{b} < \mathtt{bac} < \mathtt{bb}.$$

Conjugate words

Definition

Two words $w, w' \in \mathcal{A}^+$ are *conjugate*, denoted $w \equiv w'$, if there exist $p, s \in \mathcal{A}^+$ s.t. $w = ps$ and $w' = sp$.

The *class of conjugacy* of w is $[w] = \{w' \mid w' \equiv w\}$

Example

- $\mathrm{ab.a} \equiv \mathrm{a.ab}$, $\mathrm{a.bab} \equiv \mathrm{bab.a}$.
- $[\mathrm{aba}] = \{\mathrm{aab}, \mathrm{aba}, \mathrm{baa}\}$, $[\mathrm{abab}] = \{\mathrm{abab}, \mathrm{baba}\}$.

Lyndon words

Definition [R. Lyndon (1954), А.И. Ширшов (1953)]

$w \in \mathcal{A}^+$ is a *Lyndon word* (or *правильное слово*) if for all $p, s \in \mathcal{A}^+$ s.t. $w = ps$ one has one of the three following equivalent conditions:

1. $w < sp$,
2. $w < s$,
3. $p < s$.

Example (a < b)

a, b, ab, aab, ababb, ~~ba~~, ~~abab~~ .

Lyndon factorization

Theorem [Lyndon (1954)]

Each word $w \in \mathcal{A}^+$ can be factorized in a unique way as

$$w = \ell_1 \ell_2 \cdots \ell_n$$

with ℓ_i Lyndon word for every i and $\ell_1 \geq \ell_2 \geq \cdots \geq \ell_n$.

Example (a < b < c)

- aacab
- bc.bc.a
- b.abb.ab.a
- ab.a.a
- b.aaac.a

Lyndon factorization

Theorem [Lyndon (1954)]

Each word $w \in \mathcal{A}^+$ can be factorized in a unique way as

$$w = \ell_1 \ell_2 \cdots \ell_n$$

with ℓ_i Lyndon word for every i and $\ell_1 \geq \ell_2 \geq \cdots \geq \ell_n$.

Theorem [Duval (1980)]

The Lyndon factorization can be computed in linear time.

Lyndon factorization

Theorem [Lyndon (1954)]

Each word $w \in \mathcal{A}^+$ can be factorized in a unique way as

$$w = \ell_1 \ell_2 \cdots \ell_n$$

with ℓ_i Lyndon word for every i and $\ell_1 \geq \ell_2 \geq \cdots \geq \ell_n$.

Theorem [Duval (1980)]

The Lyndon factorization can be computed in linear time.
Proof. [*idea of*]

(∃)
- ▷ Each word has a trivial (maybe increasing) factorization in Lyndon words: $w = a_0 a_1 \cdots a_{|w|-1}$, with $a_i \in \mathcal{A}$.
- ▷ If u, v are Lyndon words and $u < v$, then uv is also Lyndon.

(!) If $w = \ell_1 \ell_2 \cdots \ell_n$ is the Lyndon factorization of w, then
- ▷ ℓ_1 is the longest prefix which is Lyndon.

To infinity and beyond

Infinite words

Definition

- $\mathbf{w} = a_0 a_1 a_2 \cdots \in \mathcal{A}^\omega$ is an *infinite word*
- $u^\omega = uuu \cdots \in \mathcal{A}^\omega$, with $u \in \mathcal{A}^+$
- Given a total order $<$ on $\mathcal{A}$, we define the *lexicographical order* on $\mathcal{A}^\omega$ as

$$\mathbf{u} < \mathbf{v} \iff \mathbf{u} = pa\mathbf{s},\ \mathbf{v} = pb\mathbf{t} \text{ with } p \in \mathcal{A}^*,\ a, b \in \mathcal{A},\ a < b,\ \mathbf{s}, \mathbf{t} \in \mathcal{A}^\omega$$

Infinite words

Definition

- $\mathbf{w} = a_0 a_1 a_2 \cdots \in \mathcal{A}^\omega$ is an *infinite word*
- $u^\omega = uuu \cdots \in \mathcal{A}^\omega$, with $u \in \mathcal{A}^+$
- Given a total order $<$ on $\mathcal{A}$, we define the *lexicographical order* on $\mathcal{A}^\omega$ as

$$\mathbf{u} < \mathbf{v} \quad \Longleftrightarrow \quad \mathbf{u} = pa\mathbf{s},\ \mathbf{v} = pb\mathbf{t} \text{ with } p \in \mathcal{A}^*,\ \ a, b \in \mathcal{A},\ \ a < b,\ \ \mathbf{s}, \mathbf{t} \in \mathcal{A}^\omega$$

When $|u| = |v|$ one has $u < v \Leftrightarrow u^\omega < v^\omega$. In general, this is not true.

Example (a < b)

$\mathtt{ab} < \mathtt{aba}$ but $(\mathtt{ab})^\omega > (\mathtt{aba})^\omega$.

$u^\omega = v^\omega \iff u$ and v are power of a common word ($\iff uv = vu$).

Infinite words

Theorem [D., Restivo, Reutenauer (2018)]

The following conditions are equivalent for nonempty words $u, v \in A^+$.

(1) $u^\omega < v^\omega$,

(2) $(uv)^\omega < v^\omega$,

(3) $u^\omega < (vu)^\omega$,

(4) $(uv)^\omega < (vu)^\omega$,

(5) $u^\omega < (uv)^\omega$,

(6) $(vu)^\omega < v^\omega$.

Infinite words

Theorem [D., Restivo, Reutenauer (2018)]

The following conditions are equivalent for nonempty words $u, v \in A^+$.

(1) $u^\omega < v^\omega$,

(2) $(uv)^\omega < v^\omega$,

(3) $u^\omega < (vu)^\omega$,

(4) $(uv)^\omega < (vu)^\omega$,

(5) $u^\omega < (uv)^\omega$,

(6) $(vu)^\omega < v^\omega$.

Examples ($\mathtt{a} < \mathtt{b}$)

Let $u = \mathtt{a}$ and $v = \mathtt{ab}$. Then

$$\mathtt{a}^\omega < (\mathtt{a.ab})^\omega < (\mathtt{ab.a})^\omega < (\mathtt{ab})^\omega$$

Infinite words

Theorem [D., Restivo, Reutenauer (2018)]

The following conditions are equivalent for nonempty words $u, v \in A^+$.

(1) $u^\omega < v^\omega$,

(2) $(uv)^\omega < v^\omega$,

(3) $u^\omega < (vu)^\omega$,

(4) $(uv)^\omega < (vu)^\omega$,

(5) $u^\omega < (uv)^\omega$,

(6) $(vu)^\omega < v^\omega$.

Examples ($\mathtt{a} < \mathtt{b}$)

Let $u = \mathtt{a}$ and $v = \mathtt{ab}$. Then

$$\mathtt{a}^\omega < (\mathtt{a.ab})^\omega < (\mathtt{ab.a})^\omega < (\mathtt{ab})^\omega$$

Infinite words

Theorem [D., Restivo, Reutenauer (2018)]

The following conditions are equivalent for nonempty words $u, v \in A^+$.

(1) $u^\omega < v^\omega$,

(2) $(uv)^\omega < v^\omega$,

(3) $u^\omega < (vu)^\omega$,

(4) $(uv)^\omega < (vu)^\omega$,

(5) $u^\omega < (uv)^\omega$,

(6) $(vu)^\omega < v^\omega$.

Examples ($\mathtt{a} < \mathtt{b}$)

Let $u = \mathtt{a}$ and $v = \mathtt{ab}$. Then

$$\mathtt{a}^\omega < (\mathtt{a.ab})^\omega < (\mathtt{ab.a})^\omega < (\mathtt{ab})^\omega$$

Infinite words

Theorem [D., Restivo, Reutenauer (2018)]

The following conditions are equivalent for nonempty words $u, v \in A^+$.

(1) $u^\omega < v^\omega$,

(2) $(uv)^\omega < v^\omega$,

(3) $u^\omega < (vu)^\omega$,

(4) $(uv)^\omega < (vu)^\omega$,

(5) $u^\omega < (uv)^\omega$,

(6) $(vu)^\omega < v^\omega$.

Examples (a < b)

Let $u = \mathtt{a}$ and $v = \mathtt{ab}$. Then

$$\mathtt{a}^\omega < (\mathtt{a.ab})^\omega < (\mathtt{ab.a})^\omega < (\mathtt{ab})^\omega$$

Infinite words

Theorem [D., Restivo, Reutenauer (2018)]

The following conditions are equivalent for nonempty words $u, v \in A^+$.

(1) $u^\omega < v^\omega$,

(2) $(uv)^\omega < v^\omega$,

(3) $u^\omega < (vu)^\omega$,

(4) $(uv)^\omega < (vu)^\omega$,

(5) $u^\omega < (uv)^\omega$,

(6) $(vu)^\omega < v^\omega$.

Examples (a < b)

Let $u = \mathtt{a}$ and $v = \mathtt{ab}$. Then

$$\mathtt{a}^\omega < (\mathtt{a.ab})^\omega < (\mathtt{ab.a})^\omega < (\mathtt{ab})^\omega$$

Infinite words

Theorem [D., Restivo, Reutenauer (2018)]

The following conditions are equivalent for nonempty words $u, v \in A^+$.

(1) $u^\omega < v^\omega$,

(2) $(uv)^\omega < v^\omega$,

(3) $u^\omega < (vu)^\omega$,

(4) $(uv)^\omega < (vu)^\omega$,

(5) $u^\omega < (uv)^\omega$,

(6) $(vu)^\omega < v^\omega$.

Examples ($\mathtt{a} < \mathtt{b}$)

Let $u = \mathtt{a}$ and $v = \mathtt{ab}$. Then

$$\mathtt{a}^\omega < (\mathtt{a.ab})^\omega < (\mathtt{ab.a})^\omega < (\mathtt{ab})^\omega$$

Lyndon words and infinite words

Corollary

Let $u, v \in \mathcal{A}^+$. Then $u^\omega < v^\omega$ if and only if $uv < vu$.

Lyndon words and infinite words

Corollary

Let $u, v \in \mathcal{A}^+$. Then $u^\omega < v^\omega$ if and only if $uv < vu$.

Theorem [D., Restivo, Reutenauer (2018)]

A word w is Lyndon iff for any non-trivial factorization $w = ps$ one of the following equivalent conditions is satisfied:

(1) $p^\omega < s^\omega$,

(2) $(ps)^\omega < s^\omega$,

(3) $p^\omega < (sp)^\omega$,

(4) $(ps)^\omega < (sp)^\omega$,

(5) $p^\omega < (ps)^\omega$ [Ufnarovskij (1995)],

(6) $(sp)^\omega < s^\omega$.

Factorization into Lyndon words

Theorem [Ufnarovskij (1995)]

Let $w = \ell_1 \ell_2 \cdots \ell_n$ the unique non-increasing factorization of w in Lyndon word. Then

- $\ell_1^\omega > (\ell_2 \cdots \ell_n)^\omega$
- ℓ_1 is the shortest nontrivial prefix p s.t. $w = ps$ and $p^\omega \geq s^\omega$,
- ℓ_1 is the shortest nontrivial prefix p s.t. $w = ps$ and $p^\omega \geq w^\omega$.

Example ($a < b$)

$w = aab.aab.a$

- $(aab)^\omega > (aab.a)^\omega$,
- $(aab)^\omega > (aab.aab.a)^\omega$.

Generalized lexicographical order

$(\mathcal{A}, <)$

$$u < v \;:\Leftrightarrow\; v = us \text{ or } \begin{cases} u = pau' \\ v = pbv' \\ a < b \end{cases}$$

$(\mathcal{A}, <_g := (<_n)_{n\in\mathbb{N}})$

$$u <_g v \;:\Leftrightarrow\; v = us \text{ or } \begin{cases} u = pau' \\ v = pbv' \\ a <_{|p|+1} b \end{cases}$$

Generalized lexicographical order

$(\mathcal{A}, <)$

$$u < v \;:\Leftrightarrow\; v = us \text{ or } \begin{cases} u = pau' \\ v = pbv' \\ a < b \end{cases}$$

$(\mathcal{A}, <_g := (<_n)_{n\in\mathbb{N}})$

$$u <_g v \;:\Leftrightarrow\; v = us \text{ or } \begin{cases} u = pau' \\ v = pbv' \\ a <_{|p|+1} b \end{cases}$$

Examples

- Classical order ($<$): $\mathtt{a} <_n \mathtt{b}$ for all $n \geq 1$.

$$\mathtt{a} < \mathtt{aa} < \mathtt{ab} < \mathtt{aba} < \mathtt{baa} < \mathtt{bab}$$

Generalized lexicographical order

$(\mathcal{A}, <)$

$$u < v \ :\Leftrightarrow\ v = us \text{ or } \begin{cases} u = pau' \\ v = pbv' \\ a < b \end{cases}$$

$(\mathcal{A}, <_g := (<_n)_{n\in\mathbb{N}})$

$$u <_g v \ :\Leftrightarrow\ v = us \text{ or } \begin{cases} u = pau' \\ v = pbv' \\ a <_{|p|+1} b \end{cases}$$

Examples

- Classical order ($<$): $\mathtt{a} <_n \mathtt{b}$ for all $n \geq 1$.
- Alternate order ($<_{alt}$): $\begin{cases} \mathtt{a} <_n \mathtt{b} & \text{if } n \equiv 1 \pmod 2 \\ \mathtt{b} <_n \mathtt{a} & \text{otherwise.} \end{cases}$

$$\mathtt{a} <_{alt} \mathtt{ab} <_{alt} \mathtt{aa} <_{alt} \mathtt{b} <_{alt} \mathtt{bba} <_{alt} \mathtt{ba}$$

Generalized lexicographical order

$(\mathcal{A}, <)$

$$u < v \;:\Leftrightarrow\; v = us \text{ or } \begin{cases} u = pau' \\ v = pbv' \\ a < b \end{cases}$$

$(\mathcal{A}, <_g := (<_n)_{n\in\mathbb{N}})$

$$u <_g v \;:\Leftrightarrow\; v = us \text{ or } \begin{cases} u = pau' \\ v = pbv' \\ a <_{|p|+1} b \end{cases}$$

Examples

- Classical order ($<$): $\mathtt{a} <_n \mathtt{b}$ for all $n \geq 1$.
- Alternate order ($<_{alt}$): $\begin{cases} \mathtt{a} <_n \mathtt{b} & \text{if } n \equiv 1 \pmod 2 \\ \mathtt{b} <_n \mathtt{a} & \text{otherwise.} \end{cases}$

$$a_1\ a_2\ a_3 \cdots <_{alt} b_1\ b_2\ b_3 \cdots \quad \Longleftrightarrow \quad a_1 + \cfrac{1}{a_2 + \cfrac{1}{a_3 + \cfrac{1}{\cdots}}} < b_1 + \cfrac{1}{b_2 + \cfrac{1}{b_3 + \cfrac{1}{\cdots}}}$$

Generalized lexicographical order

$(\mathcal{A}, <)$

$$u < v \quad :\Leftrightarrow \quad v = us \text{ or } \begin{cases} u = pau' \\ v = pbv' \\ a < b \end{cases}$$

$(\mathcal{A}, <_g := (<_n)_{n\in\mathbb{N}})$

$$u <_g v \quad :\Leftrightarrow \quad v = us \text{ or } \begin{cases} u = pau' \\ v = pbv' \\ a <_{|p|+1} b \end{cases}$$

Examples

- Classical order ($<$): $\mathtt{a} <_n \mathtt{b}$ for all $n \geq 1$.
- Alternate order ($<_{alt}$): $\begin{cases} \mathtt{a} <_n \mathtt{b} & \text{if } n \equiv 1 \pmod 2 \\ \mathtt{b} <_n \mathtt{a} & \text{otherwise.} \end{cases}$
- Prime order ($<_\pi$) : $\begin{cases} \mathtt{b} <_n \mathtt{a} & \text{if } n \text{ is prime} \\ \mathtt{a} <_n \mathtt{b} & \text{otherwise.} \end{cases}$

$$\mathtt{aba} <_\pi \mathtt{abaa} <_\pi \mathtt{aab} <_\pi \mathtt{bab} <_\pi \mathtt{baab}$$

Generalized Lyndon words

A word $w \in \mathcal{A}^+$ is a *(classical) Lyndon word* if for any nontrivial factorization $w = ps$ one has $w < sp$.

Generalized Lyndon words

A word $w \in \mathcal{A}^+$ is a *(classical) Lyndon word* if for any nontrivial factorization $w = ps$ one has $w^\omega < (sp)^\omega$.

Generalized Lyndon words

A word $w \in \mathcal{A}^+$ is a *generalized Lyndon word* if for any nontrivial factorization $w = ps$ one has $w^\omega <_g (sp)^\omega$.

Generalized Lyndon words

A word $w \in \mathcal{A}^+$ is a *generalized Lyndon word* if for any nontrivial factorization $w = ps$ one has $w^\omega <_g (sp)^\omega$.

Theorem [Reutenauer (2005), D., Restivo, Reutenauer (2018)]

A word w is a generalized Lyndon word if and only if for any non trivial factorization $w = ps$ one has $p^\omega <_g s^\omega$ and $w^\omega <_g s^\omega$.

Generalized Lyndon words

A word $w \in \mathcal{A}^+$ is a *generalized Lyndon word* if for any nontrivial factorization $w = ps$ one has $w^\omega <_g (sp)^\omega$.

Theorem [Reutenauer (2005), D., Restivo, Reutenauer (2018)]

A word w is a generalized Lyndon word if and only if for any non trivial factorization $w = ps$ one has $p^\omega <_g s^\omega$ and $w^\omega <_g s^\omega$.

Theorem [Reutenauer (2005), D., Restivo, Reutenauer (2018)]

Each word $w \in \mathcal{A}^+$ can be factorized in a unique way as $w = \ell_1 \ell_2 \cdots \ell_n$, with ℓ_i generalized Lyndon words s.t. $\ell_1^\omega \geq_g \ell_2^\omega \geq_g \cdots \geq_g \ell_n^\omega$.
Moreover, ℓ_n is

- the shortest suffix s of w s.t. s^ω is minimum,
- the longest suffix of w which is a generalized Lyndon word.

Factorization of infinite words

An infinite word $\mathbf{w} \in \mathcal{A}^\omega$ is a (generalized) *Lyndon* word if $\mathbf{w} < \mathbf{s}$ for each suffix $\mathbf{s}$ of $\mathbf{w}$.

Factorization of infinite words

An infinite word $\mathbf{w} \in \mathcal{A}^\omega$ is a (generalized) *Lyndon* word if $\mathbf{w} < \mathbf{s}$ for each suffix $\mathbf{s}$ of $\mathbf{w}$.

Example ($a < b$)

- The word ab^ω is a classical Lyndon word.
- The word aba^ω is a Galois word.

Factorization of infinite words

An infinite word $\mathbf{w} \in \mathcal{A}^\omega$ is a (generalized) *Lyndon* word if $\mathbf{w} < \mathbf{s}$ for each suffix $\mathbf{s}$ of $\mathbf{w}$.

Theorem [Siromoney, Mathew, Dare, Subramanian (2005)]

Each infinite word $\mathbf{w} \in \mathcal{A}^\omega$ can be factorized in a unique was either as

- an infinite product of non-increasing finite classical Lyndon words

$$\mathbf{w} = \ell_1 \ell_2 \ell_3 \cdots \quad \text{with} \quad \ell_1 \geq \ell_2 \geq \ell_3 \geq \cdots$$

- a finite product of non-increasing classical Lyndon words

$$\mathbf{w} = \ell_1 \cdots \ell_{n-1} \boldsymbol{\ell}_\mathbf{n} \quad \text{with} \quad \ell_1 \geq \cdots \geq \ell_{n-1} \geq \ell_n, \quad |\ell_1 \cdots \ell_n| < \infty \quad \text{and} \quad \boldsymbol{\ell}_\mathbf{n} \ \text{infinite}.$$

Factorization of infinite words

An infinite word $\mathbf{w} \in \mathcal{A}^\omega$ is a (generalized) *Lyndon* word if $\mathbf{w} < \mathbf{s}$ for each suffix $\mathbf{s}$ of $\mathbf{w}$.

Theorem [Burcroff, Winsor (2019); Postic, Zamboni (2019)]

Let $<_g$ be a generalized order.
Each infinite word $\mathbf{w} \in \mathcal{A}^\omega$ can be factorized in a unique was either as

- an infinite product of non-increasing finite classical Lyndon words

$$\mathbf{w} = \ell_1 \ell_2 \ell_3 \cdots \quad \text{with} \quad \ell_1^\omega \geq_g \ell_2^\omega \geq_g \ell_3 \geq_g \cdots$$

- a finite product of non-increasing classical Lyndon words

$$\mathbf{w} = \ell_1 \cdots \ell_{n-1} \boldsymbol{\ell_n} \quad \text{with} \quad \ell_1^\omega \geq_g \cdots \geq_g \ell_{n-1}^\omega \geq_g \boldsymbol{\ell_n}, \quad |\ell_1 \cdots \ell_n| < \infty, \quad \boldsymbol{\ell_n} \text{ infinite}.$$

Complete trees

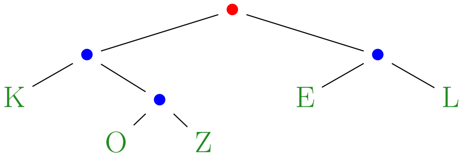
The set of *complete binary trees* over $\mathcal{A}$ is defined recursively as follows:

- each letter $a \in \mathcal{A}$ is a tree;
- if $\mathfrak{t}_1, \mathfrak{t}_2$ are trees, then $(\mathfrak{t}_1, \mathfrak{t}_2)$ is a tree.

Complete trees

The set of *complete binary trees* over $\mathcal{A}$ is defined recursively as follows:

- each letter $a \in \mathcal{A}$ is a tree;
- if $\mathfrak{t}_1, \mathfrak{t}_2$ are trees, then $(\mathfrak{t}_1, \mathfrak{t}_2)$ is a tree.



We will use the classical notions of *root*, *internal node* and *leaf* for a tree.
The *foliage* $\varphi(\mathfrak{t})$ of a tree $\mathfrak{t}$ is defined as:

- $\varphi(a) = a$ for any $a \in \mathcal{A}$,
- $\varphi((\mathfrak{t}_1, \mathfrak{t}_2)) = \varphi(\mathfrak{t}_1)\varphi(\mathfrak{t}_2)$ for any two trees $\mathfrak{t}_1, \mathfrak{t}_2$.

Left standard factorization

Definition

Let w be a Lyndon word of length at least 2.
The *left standard factorization* of w is the factorization $w = ps$, where p is the longest nonempty proper prefix of w which is a Lyndon word.

Example ($a < b < c$)

The left standard factorization of aabaacab is aabaac.ab.

Left standard factorization

Definition

Let w be a Lyndon word of length at least 2.
The *left standard factorization* of w is the factorization $w = ps$, where p is the longest nonempty proper prefix of w which is a Lyndon word.

Proposition

Both p and s are Lyndon words.
Moreover, either s is a letter or $s = s_1 s_2$, and $s_1 \leq p$.

Example ($\mathtt{a} < \mathtt{b} < \mathtt{c}$)

The left standard factorization of aabaacab is aabaac.ab.
The left standard factorization of ab is a.b, and a $\leq$ aabaac.

Left Lyndon tree

Let $w \in \mathcal{A}^+$ be a Lyndon word. Its *left Lyndon tree* $\mathcal{L}(w)$ is defined as:

- $\mathcal{L}(a) = a$ for each letter $a \in \mathcal{A}$;
- $\mathcal{L}(w) = (\mathcal{L}(p), \mathcal{L}(s))$ for each Lyndon word $w \notin \mathcal{A}$ with left standard factorization $w = ps$.

Left Lyndon tree

Let $w \in \mathcal{A}^+$ be a Lyndon word. Its *left Lyndon tree* $\mathcal{L}(w)$ is defined as:

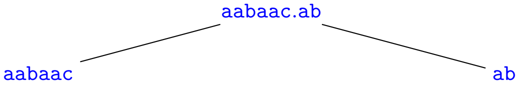
- $\mathcal{L}(a) = a$ for each letter $a \in \mathcal{A}$;
- $\mathcal{L}(w) = (\mathcal{L}(p), \mathcal{L}(s))$ for each Lyndon word $w \notin \mathcal{A}$ with left standard factorization $w = ps$.

aabaacab

Left Lyndon tree

Let $w \in \mathcal{A}^+$ be a Lyndon word. Its *left Lyndon tree* $\mathcal{L}(w)$ is defined as:

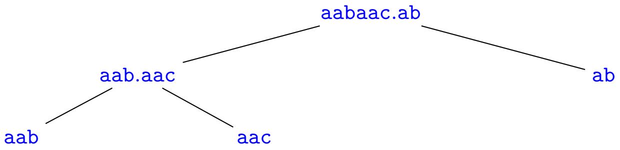
- $\mathcal{L}(a) = a$ for each letter $a \in \mathcal{A}$;
- $\mathcal{L}(w) = (\mathcal{L}(p), \mathcal{L}(s))$ for each Lyndon word $w \notin \mathcal{A}$ with left standard factorization $w = ps$.



Left Lyndon tree

Let $w \in \mathcal{A}^+$ be a Lyndon word. Its *left Lyndon tree* $\mathcal{L}(w)$ is defined as:

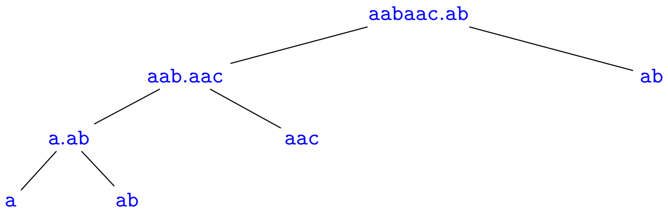
- $\mathcal{L}(a) = a$ for each letter $a \in \mathcal{A}$;
- $\mathcal{L}(w) = (\mathcal{L}(p), \mathcal{L}(s))$ for each Lyndon word $w \notin \mathcal{A}$ with left standard factorization $w = ps$.



Left Lyndon tree

Let $w \in \mathcal{A}^+$ be a Lyndon word. Its *left Lyndon tree* $\mathcal{L}(w)$ is defined as:

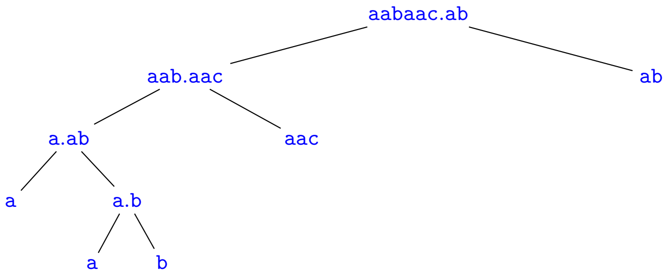
- $\mathcal{L}(a) = a$ for each letter $a \in \mathcal{A}$;
- $\mathcal{L}(w) = (\mathcal{L}(p), \mathcal{L}(s))$ for each Lyndon word $w \notin \mathcal{A}$ with left standard factorization $w = ps$.



Left Lyndon tree

Let $w \in \mathcal{A}^+$ be a Lyndon word. Its *left Lyndon tree* $\mathcal{L}(w)$ is defined as:

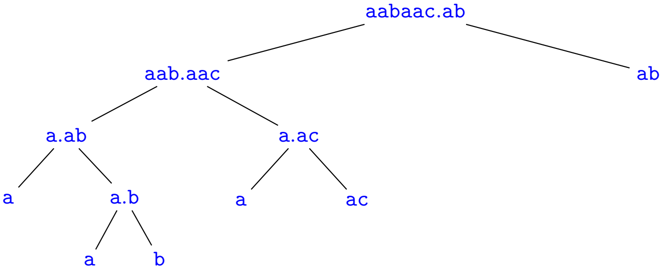
- $\mathcal{L}(a) = a$ for each letter $a \in \mathcal{A}$;
- $\mathcal{L}(w) = (\mathcal{L}(p), \mathcal{L}(s))$ for each Lyndon word $w \notin \mathcal{A}$ with left standard factorization $w = ps$.



Left Lyndon tree

Let $w \in \mathcal{A}^+$ be a Lyndon word. Its *left Lyndon tree* $\mathcal{L}(w)$ is defined as:

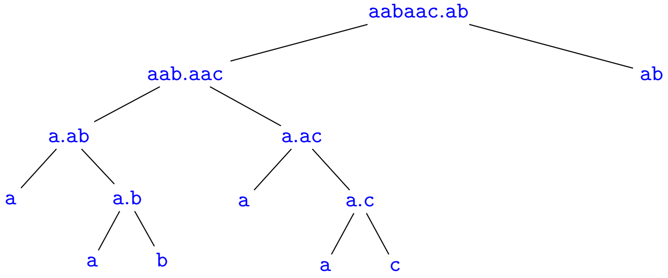
- $\mathcal{L}(a) = a$ for each letter $a \in \mathcal{A}$;
- $\mathcal{L}(w) = (\mathcal{L}(p), \mathcal{L}(s))$ for each Lyndon word $w \notin \mathcal{A}$ with left standard factorization $w = ps$.



Left Lyndon tree

Let $w \in \mathcal{A}^+$ be a Lyndon word. Its *left Lyndon tree* $\mathcal{L}(w)$ is defined as:

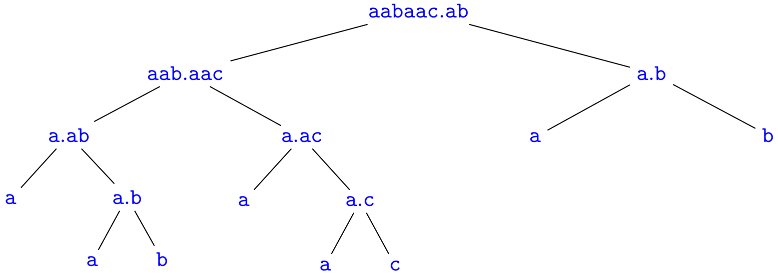
- $\mathcal{L}(a) = a$ for each letter $a \in \mathcal{A}$;
- $\mathcal{L}(w) = (\mathcal{L}(p), \mathcal{L}(s))$ for each Lyndon word $w \notin \mathcal{A}$ with left standard factorization $w = ps$.



Left Lyndon tree

Let $w \in \mathcal{A}^+$ be a Lyndon word. Its *left Lyndon tree* $\mathcal{L}(w)$ is defined as:

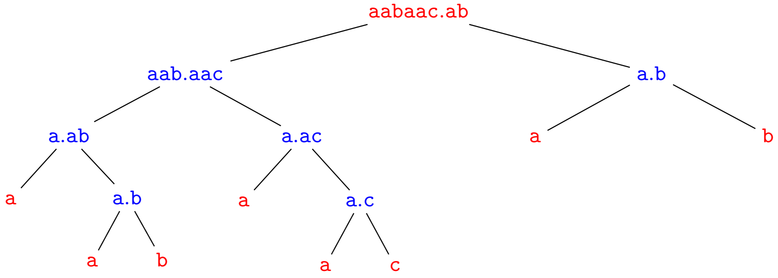
- $\mathcal{L}(a) = a$ for each letter $a \in \mathcal{A}$;
- $\mathcal{L}(w) = (\mathcal{L}(p), \mathcal{L}(s))$ for each Lyndon word $w \notin \mathcal{A}$ with left standard factorization $w = ps$.



Left Lyndon tree

Let $w \in \mathcal{A}^+$ be a Lyndon word. Its *left Lyndon tree* $\mathcal{L}(w)$ is defined as:

- $\mathcal{L}(a) = a$ for each letter $a \in \mathcal{A}$;
- $\mathcal{L}(w) = (\mathcal{L}(p), \mathcal{L}(s))$ for each Lyndon word $w \notin \mathcal{A}$ with left standard factorization $w = ps$.



Clearly $\varphi(\mathcal{L}(w)) = w$.

Prefix standardization

$$u \prec v \quad :\Longleftrightarrow \quad \begin{cases} u^\omega < v^\omega & \text{or} \\ u^\omega = v^\omega & \text{and } |u| > |v|. \end{cases}$$

Example (a < b)

aa ≺ a ≺ ab ≺ ba ≺ b.

Prefix standardization

$$u \prec v \quad :\Longleftrightarrow \quad \begin{cases} u^\omega < v^\omega & \text{or} \\ u^\omega = v^\omega & \text{and } |u| > |v|. \end{cases}$$

Example ($a < b$)

aa $\prec$ a $\prec$ ab $\prec$ ba $\prec$ b.

The *prefix standard permutation* associated to a word w is obtained by ordering the nonempty prefixes of w according to $\prec$.

Prefix standardization

$$u \prec v \quad :\Longleftrightarrow \quad \begin{cases} u^\omega < v^\omega & \text{or} \\ u^\omega = v^\omega & \text{and } |u| > |v|. \end{cases}$$

Example (a < b)

aa $\prec$ a $\prec$ ab $\prec$ ba $\prec$ b.

The *prefix standard permutation* associated to a word w is obtained by ordering the nonempty prefixes of w according to $\prec$.

Example (a < b)

$$w \quad = \quad \texttt{aabaacab}$$

Prefix standardization

$$u \prec v \quad :\Longleftrightarrow \quad \begin{cases} u^\omega < v^\omega & \text{or} \\ u^\omega = v^\omega & \text{and } |u|>|v|. \end{cases}$$

Example (a < b)

aa $\prec$ a $\prec$ ab $\prec$ ba $\prec$ b.

The *prefix standard permutation* associated to a word w is obtained by ordering the nonempty prefixes of w according to $\prec$.

Example (a < b)

w = aabaacab

1

aa

Prefix standardization

$$u \prec v \quad :\Longleftrightarrow \quad \begin{cases} u^\omega < v^\omega & \text{or} \\ u^\omega = v^\omega & \text{and } |u| > |v|. \end{cases}$$

Example (a < b)

aa $\prec$ a $\prec$ ab $\prec$ ba $\prec$ b.

The *prefix standard permutation* associated to a word w is obtained by ordering the nonempty prefixes of w according to $\prec$.

Example (a < b)

w = aabaacab

21

aa $\prec$ a

Prefix standardization

$$u \prec v \quad :\Longleftrightarrow \quad \begin{cases} u^\omega < v^\omega & \text{or} \\ u^\omega = v^\omega & \text{and } |u|>|v|. \end{cases}$$

Example (a < b)

aa ≺ a ≺ ab ≺ ba ≺ b.

The *prefix standard permutation* associated to a word w is obtained by ordering the nonempty prefixes of w according to $\prec$.

Example (a < b)

w = aabaacab

21 3

aa ≺ a ≺ aabaa

Prefix standardization

$$u \prec v \quad :\Longleftrightarrow \quad \begin{cases} u^\omega < v^\omega & \text{or} \\ u^\omega = v^\omega & \text{and } |u|>|v|. \end{cases}$$

Example (a < b)

aa ≺ a ≺ ab ≺ ba ≺ b.

The *prefix standard permutation* associated to a word w is obtained by ordering the nonempty prefixes of w according to $\prec$.

Example (a < b)

w = aabaacab
21 43

aa ≺ a ≺ aabaa ≺ aaba

Prefix standardization

$$u \prec v \quad :\Longleftrightarrow \quad \begin{cases} u^\omega < v^\omega & \text{or} \\ u^\omega = v^\omega & \text{and } |u|>|v|. \end{cases}$$

Example (a < b)

aa ≺ a ≺ ab ≺ ba ≺ b.

The *prefix standard permutation* associated to a word w is obtained by ordering the nonempty prefixes of w according to $\prec$.

Example (a < b)

$$w = \text{aabaacab}$$

21543

aa ≺ a ≺ aabaa ≺ aaba ≺ aab

Prefix standardization

$$u \prec v \quad :\Longleftrightarrow \quad \begin{cases} u^\omega < v^\omega & \text{or} \\ u^\omega = v^\omega & \text{and } |u|>|v|. \end{cases}$$

Example (a < b)

aa $\prec$ a $\prec$ ab $\prec$ ba $\prec$ b.

The *prefix standard permutation* associated to a word w is obtained by ordering the nonempty prefixes of w according to $\prec$.

Example (a < b)

w = aabaacab
21543 6

aa $\prec$ a $\prec$ aabaa $\prec$ aaba $\prec$ aab $\prec$ aabaaca

Prefix standardization

$$u \prec v \quad :\Longleftrightarrow \quad \begin{cases} u^\omega < v^\omega & \text{or} \\ u^\omega = v^\omega & \text{and } |u|>|v|. \end{cases}$$

Example (a < b)

aa ≺ a ≺ ab ≺ ba ≺ b.

The *prefix standard permutation* associated to a word w is obtained by ordering the nonempty prefixes of w according to $\prec$.

Example (a < b)

w = aabaacab
2154376

aa ≺ a ≺ aabaa ≺ aaba ≺ aab ≺ aabaaca ≺ aabac

Prefix standardization

$$u \prec v \quad :\Longleftrightarrow \quad \begin{cases} u^\omega < v^\omega & \text{or} \\ u^\omega = v^\omega & \text{and } |u|>|v|. \end{cases}$$

Example (a < b)

aa $\prec$ a $\prec$ ab $\prec$ ba $\prec$ b.

The *prefix standard permutation* associated to a word w is obtained by ordering the nonempty prefixes of w according to $\prec$.

Example (a < b)

w = aabaacab
21543768

aa $\prec$ a $\prec$ aabaa $\prec$ aaba $\prec$ aab $\prec$ aabaaca $\prec$ aabac $\prec$ w

Prefix standardization

$$u \prec v \quad :\Longleftrightarrow \quad \begin{cases} u^\omega < v^\omega & \text{or} \\ u^\omega = v^\omega & \text{and } |u|>|v|. \end{cases}$$

Example (a < b)

aa $\prec$ a $\prec$ ab $\prec$ ba $\prec$ b.

The *prefix standard permutation* associated to a word w is obtained by ordering the nonempty prefixes of w according to $\prec$.

Example (a < b)

w = aabaacab
21543768

aa $\prec$ a $\prec$ aabaa $\prec$ aaba $\prec$ aab $\prec$ aabaaca $\prec$ aabac $\prec$ w

Left Cartesian tree

Theorem [Ufnarovskij (1995)]

w is a Lyndon word if and only if for any nontrivial factorization $w = ps$ one has $p^\omega < w^\omega$.

Left Cartesian tree

Theorem [Ufnarovskij (1995)]

w is a Lyndon word if and only if for any nontrivial factorization $w = ps$ one has $p^\omega < w^\omega$.

`aabaacab` $\longleftrightarrow$ 2154376~~8~~

Left Cartesian tree

Theorem [Ufnarovskij (1995)]

w is a Lyndon word if and only if for any nontrivial factorization $w = ps$ one has $p^\omega < w^\omega$.

aabaacab $\longleftrightarrow$ 2154376

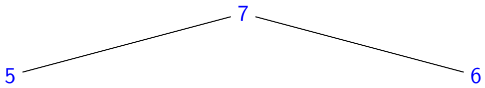
7

Left Cartesian tree

Theorem [Ufnarovskij (1995)]

w is a Lyndon word if and only if for any nontrivial factorization $w = ps$ one has $p^\omega < w^\omega$.

aabaacab $\longleftrightarrow$ 2154376

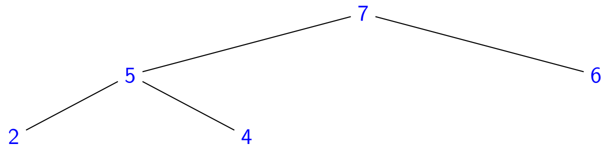


Left Cartesian tree

Theorem [Ufnarovskij (1995)]

w is a Lyndon word if and only if for any nontrivial factorization $w = ps$ one has $p^\omega < w^\omega$.

aabaacab $\longleftrightarrow$ 2154376

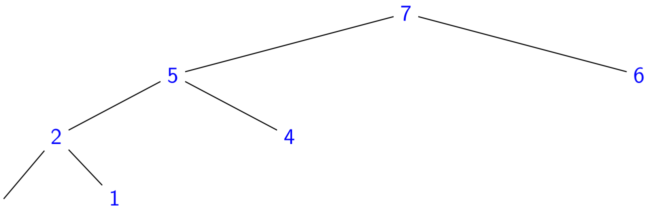


Left Cartesian tree

Theorem [Ufnarovskij (1995)]

w is a Lyndon word if and only if for any nontrivial factorization $w = ps$ one has $p^\omega < w^\omega$.

`aabaacab` $\longleftrightarrow$ 2154376

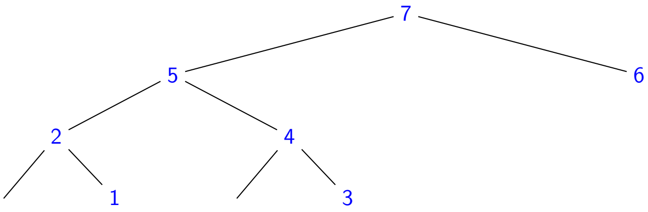


Left Cartesian tree

Theorem [Ufnarovskij (1995)]

w is a Lyndon word if and only if for any nontrivial factorization $w = ps$ one has $p^\omega < w^\omega$.

aabaacab $\longleftrightarrow$ 2154376

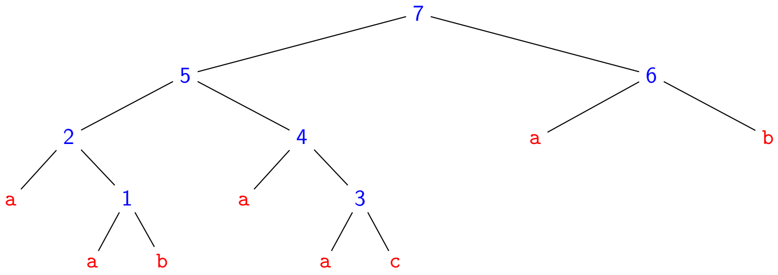


Left Cartesian tree

Theorem [Ufnarovskij (1995)]

w is a Lyndon word if and only if for any nontrivial factorization $w = ps$ one has $p^\omega < w^\omega$.

aabaacab $\longleftrightarrow$ 2154376



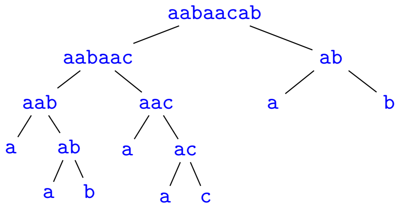
We complete in such a way that $\varphi(\mathcal{C}(w)) = w$.

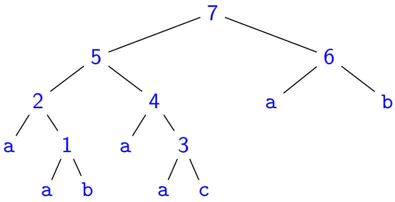
Equivalence of trees

Theorem [D., Restivo, Reutenauer (2019)]

Let w be a Lyndon word. Then $\mathcal{L}(w) = \mathcal{C}(w)$.

aabaacab ⟷ 2154376



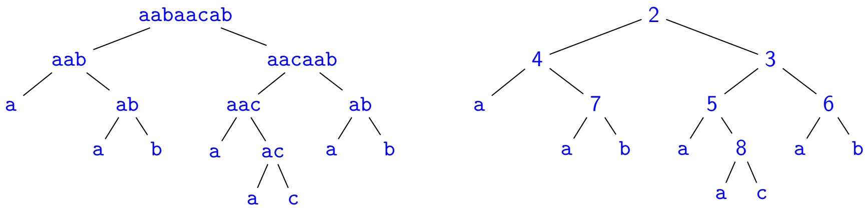


Right Lyndon tree

The *right Lyndon tree* is defined symmetrically using the *right standard factorization* $w = ps$ of a Lyndon word, where s is the longest proper Lyndon suffix of w.

Theorem [Hohlweg, Reutenauer (2003)]

Let w be a Lyndon word. Then $\mathcal{R}(w) = \mathcal{C}_R(w)$.



w = aabaacab
14725836

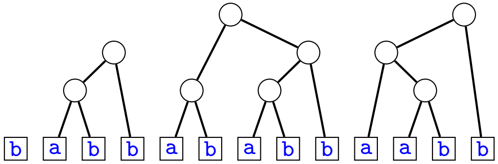
w < aacab < ab < abaacab < acab < b < baacab < cab

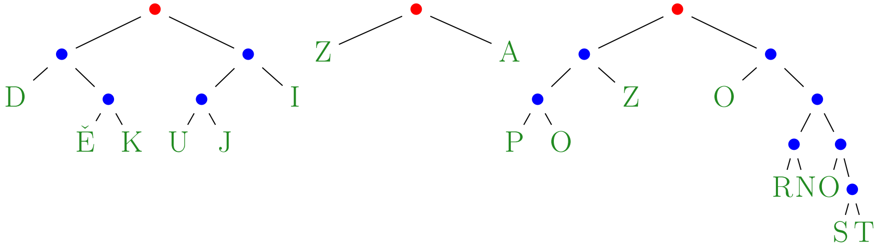
An algorithm

using Lyndon forests

Theorem [Badkobeh, Crochemore (2020)]

Sorting prefixes of a (not necessarily Lyndon) word using the order $\prec$ can be done in linear time.




D
Ě
K
U
J
I
Z
A
P
O
Z
O
R
N
O
S
T