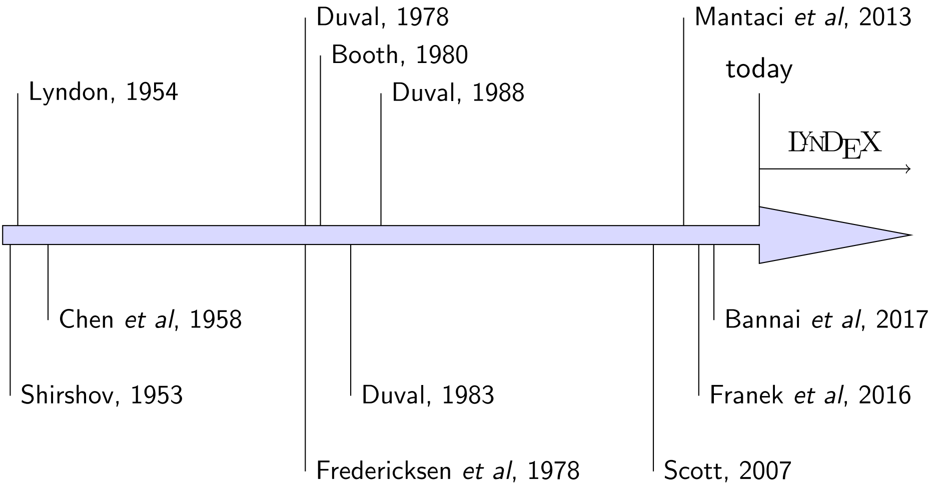
The LYNDEX Project

Arnaud Lefebvre

14 December 2023

Back to the origins. . .



Standard sequences: Lyndon, 1954

Let C_n be the set of all sequences c of length n, and define S_n to be the subset of those "standard" c that have the property of preceding lexicographically all of their own proper terminal segments $c_k c_{k+1} \cdots c_n$, $1 < k \leqq n$.

`"algorithmics"` is a standard sequence.

`"mathematics"` is not a standard sequence.

Regular words: Shirshov, 1953

Подалгебры свободных лиевых алгебр

А. И. Ширшов (Москва)

Определение 1. Слова длины 1, т. е. сами элементы множества R, назовем правильными словами и произвольно упорядочим. Считая, что правильные слова, длины которых меньше n, $n > 1$, уже определены и упорядочены при помощи отношения $\leqslant$ так, что слова меньшей длины предшествуют словам большей длины, назовем слово w длины n правильным при выполнении условий:

1) $w = uv$, где u, v — правильные слова и $u > v$;
2) если $u = u_1 u_2$, то $u_2 \leqslant v$.

Определенные таким образом правильные слова длины n произвольно упорядочим и положим, что они больше правильных слов меньшей длины.

Regular words: Shirshov, 1953

Definition 1. We will call words of length 1, i.e., elements of R, *regular words*, and we will order them arbitrarily. Assuming that regular words of length less than n, $n > 1$, are already defined and ordered by the relation $\leq$ in such a way that shorter words precede longer words, we call a word w of length n *regular* if the following conditions are satisfied:

1) $w = uv$ where u and v are regular words and $u > v$;
2) if $u = u_1 u_2$ then $u_2 \leq v$.

We will order arbitrarily the regular words of length n defined in this way, and declare that they are greater than shorter words.

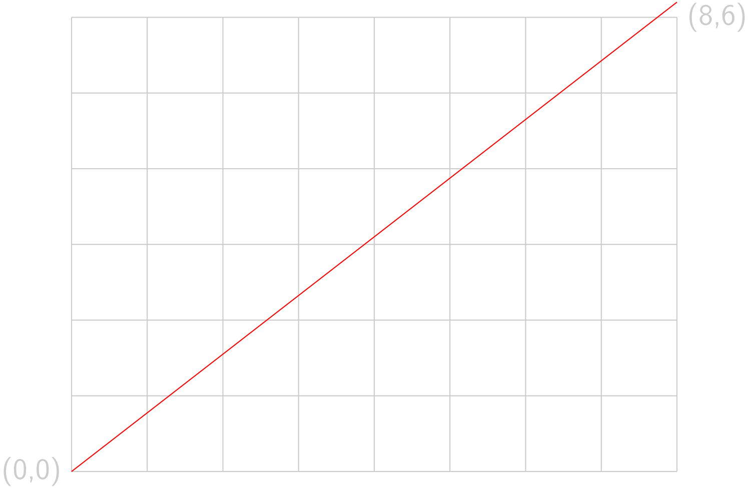
Lyndon words

Let $\mathtt{w}$ be a Lyndon word (not reduced to a single letter):

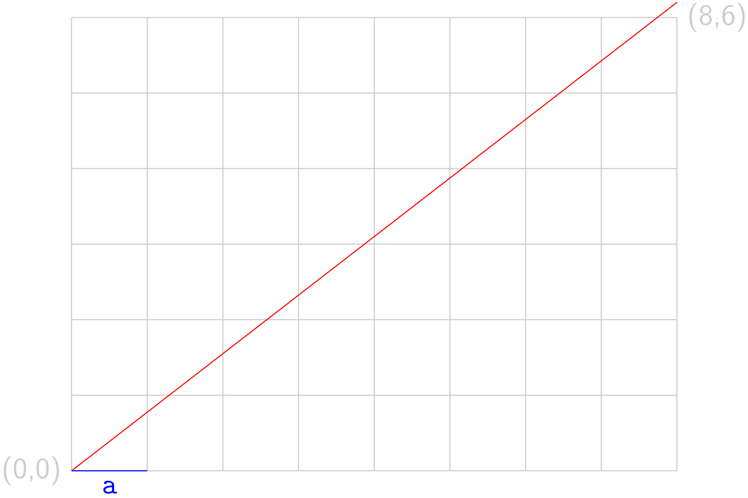
- $\mathtt{w}$ is strictly lexicographically smaller than all its proper suffixes
- $\mathtt{w}$ is the smallest element of its conjugacy class
- let $\mathtt{v}$ be the longest proper suffix of $\mathtt{w}$ that is a Lyndon word, then $\mathtt{w} = \mathtt{uv}$ where $\mathtt{u}$ is also a Lyndon word and $\mathtt{u} <_{\text{LEX}} \mathtt{v}$: it is called the "standard factorization" or "right standard factorization"
- similarly, let $\mathtt{u'}$ be the longest proper prefix of $\mathtt{w}$ that is a Lyndon word, then $\mathtt{w} = \mathtt{u'v'}$ where $\mathtt{v'}$ is also a Lyndon word and $\mathtt{u'} <_{\text{LEX}} \mathtt{v'}$: it is called the "left standard factorization"

Lyndon words: a 2D point of view

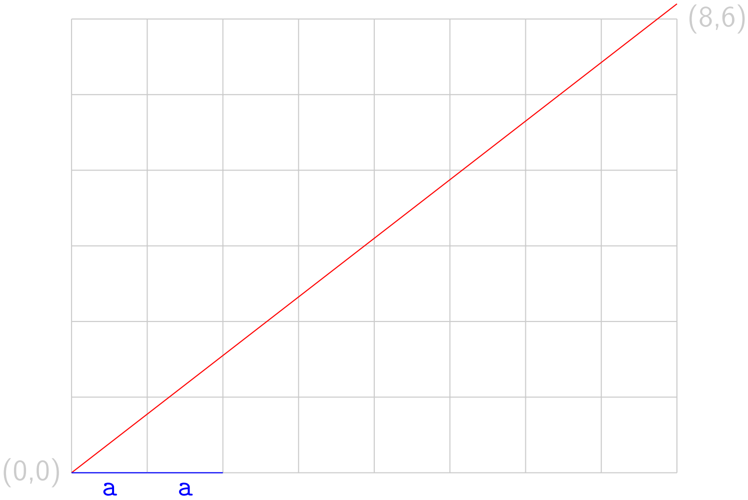
Lyndon words: a 2D point of view



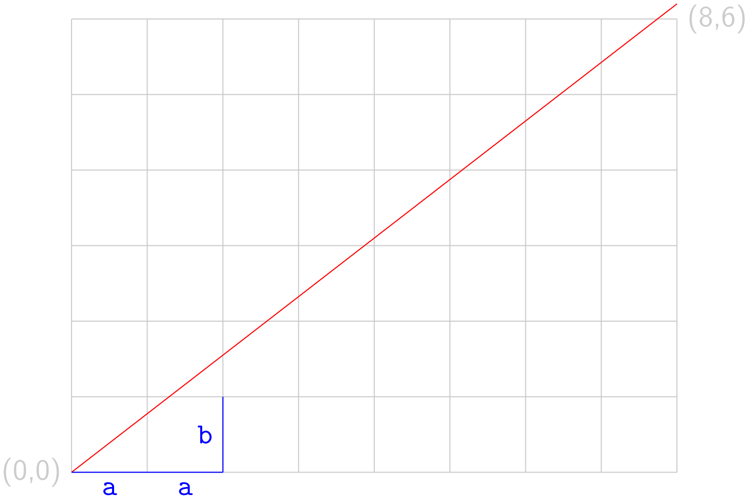
Lyndon words: a 2D point of view



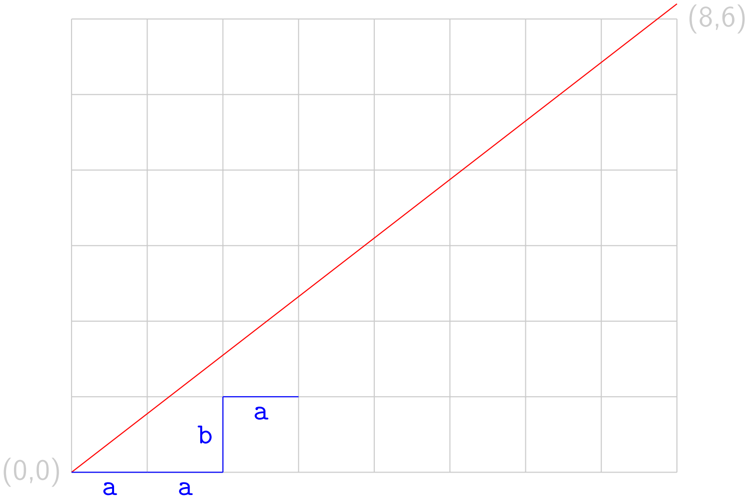
Lyndon words: a 2D point of view



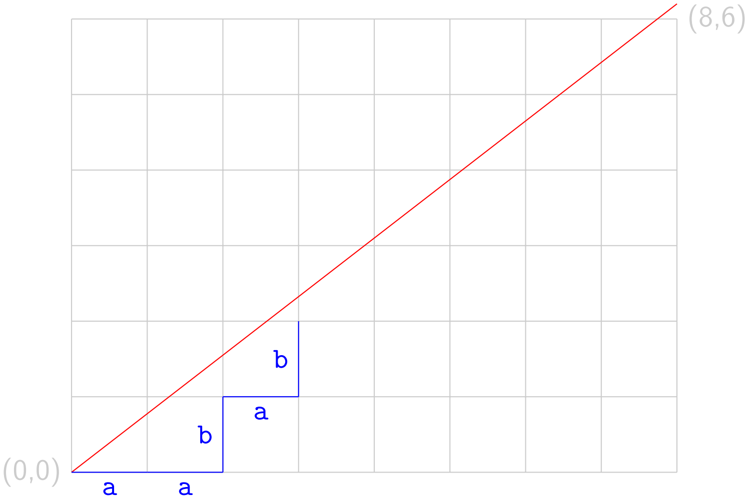
Lyndon words: a 2D point of view



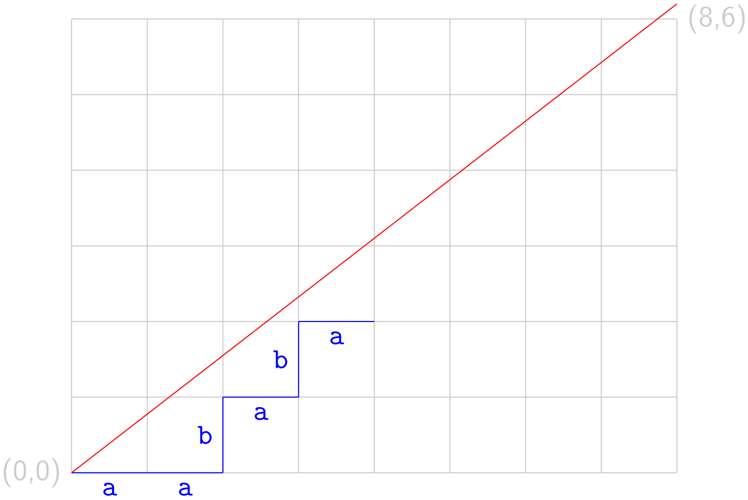
Lyndon words: a 2D point of view



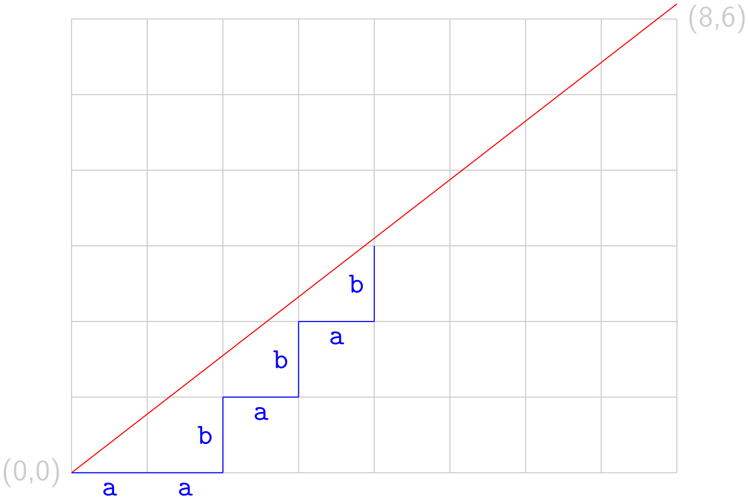
Lyndon words: a 2D point of view



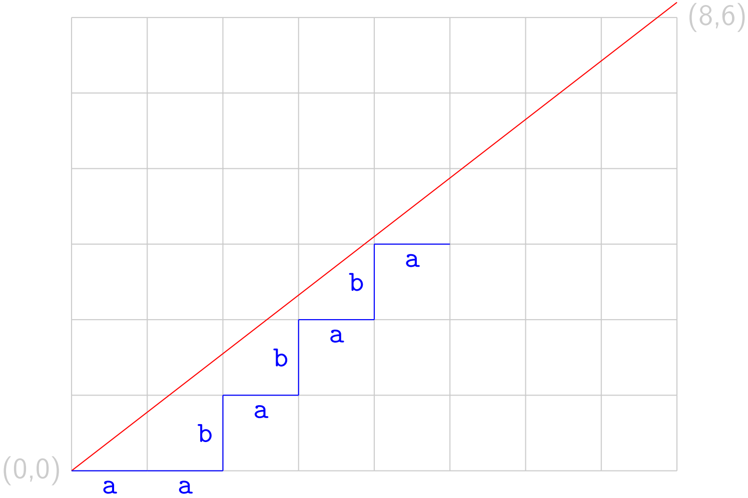
Lyndon words: a 2D point of view



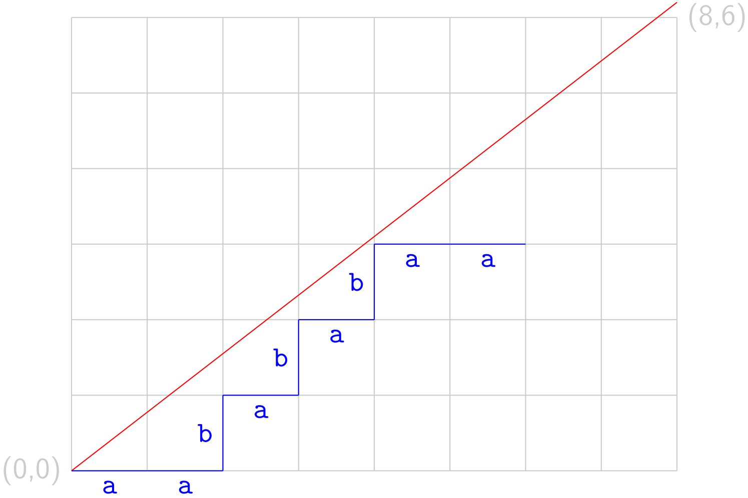
Lyndon words: a 2D point of view



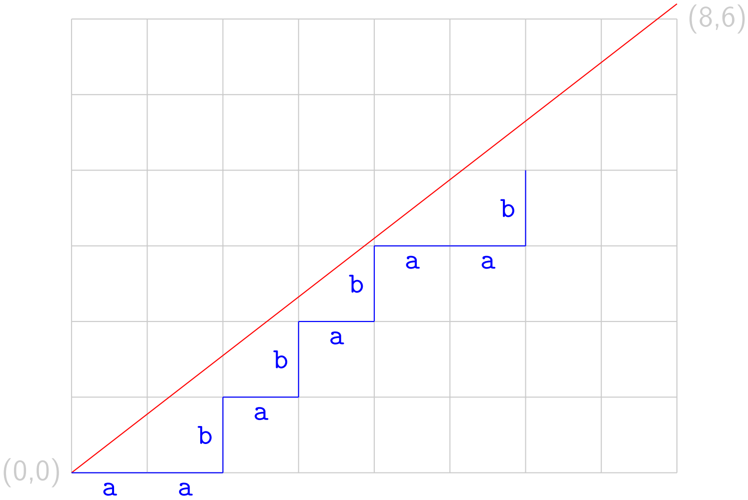
Lyndon words: a 2D point of view



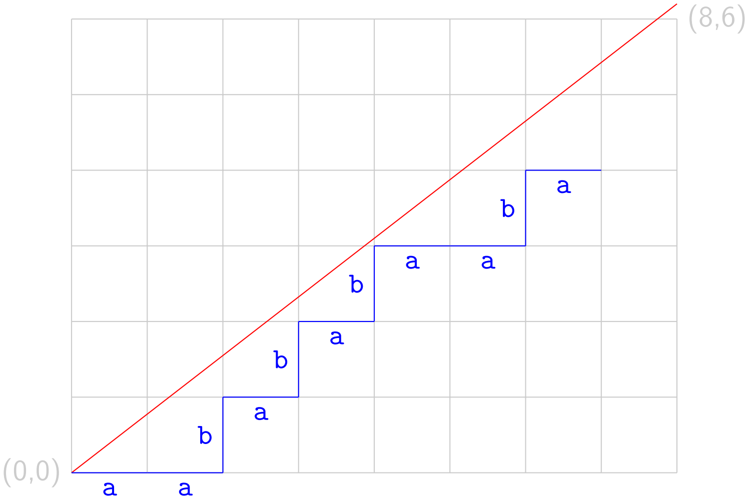
Lyndon words: a 2D point of view



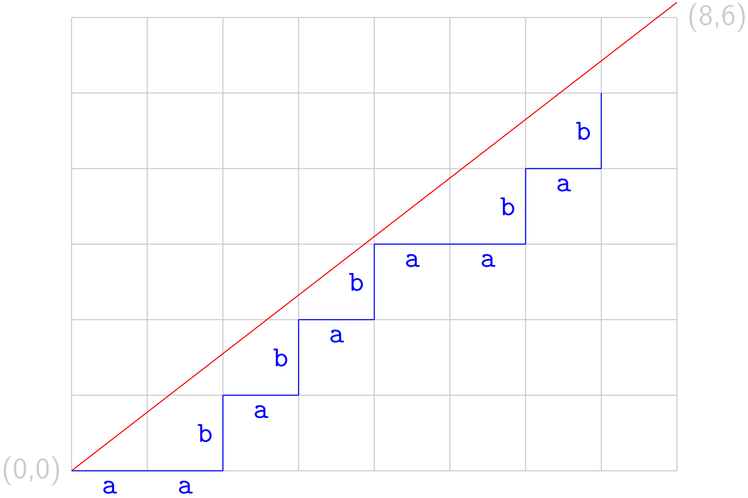
Lyndon words: a 2D point of view



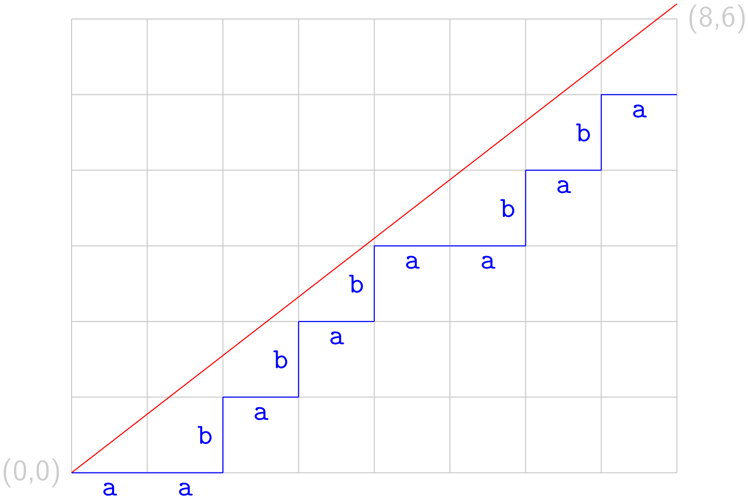
Lyndon words: a 2D point of view



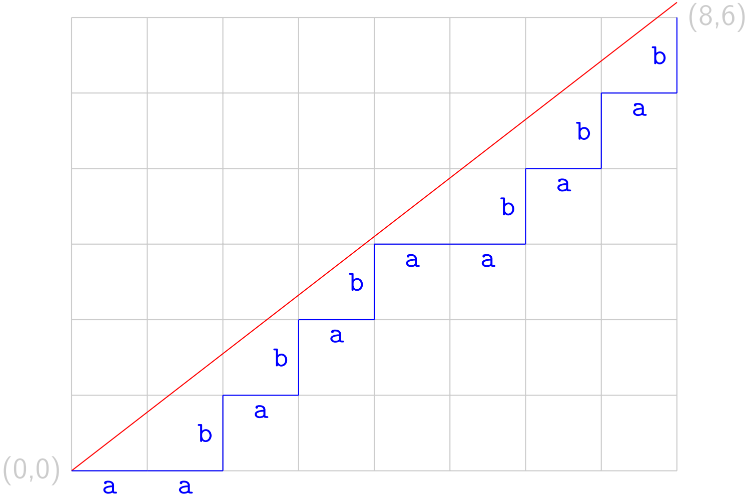
Lyndon words: a 2D point of view



Lyndon words: a 2D point of view

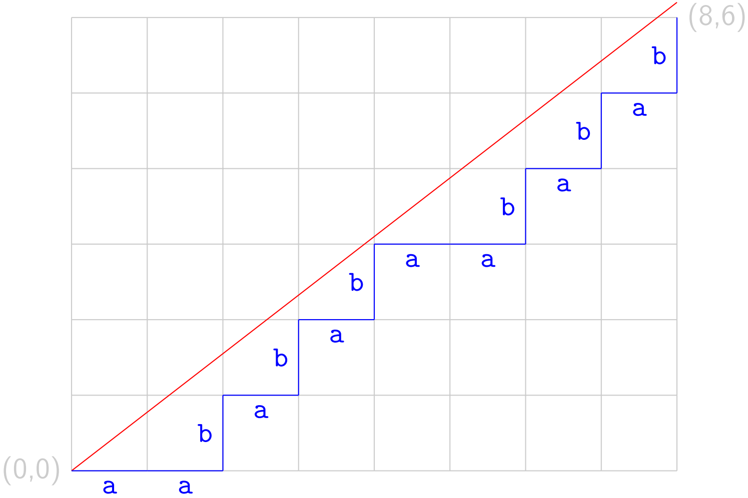


Lyndon words: a 2D point of view

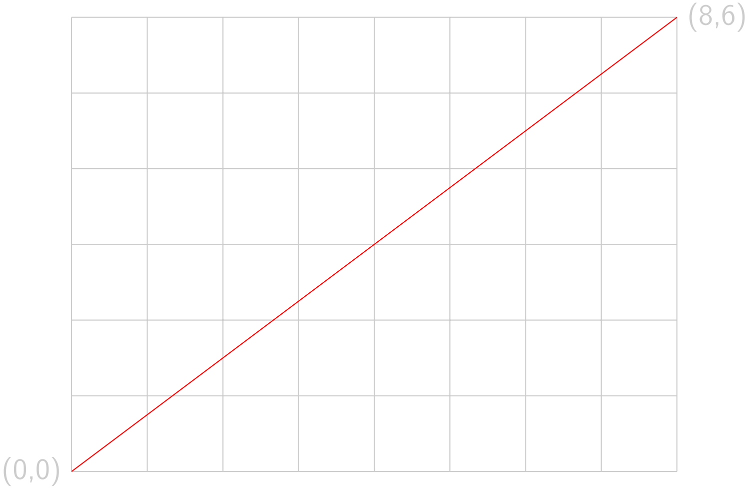


Lyndon words: a 2D point of view

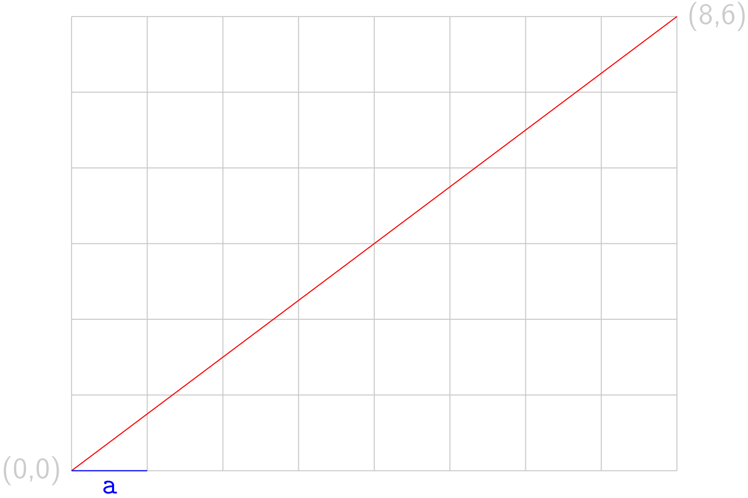
irrational slope = sturmian word



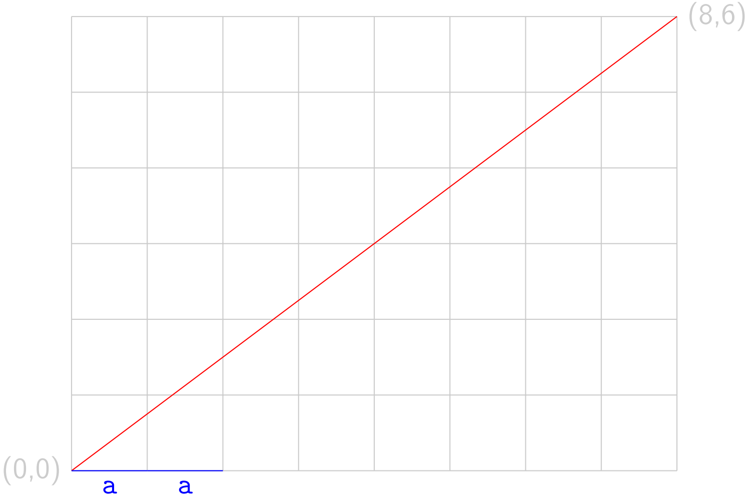
Lyndon words: a 2D point of view



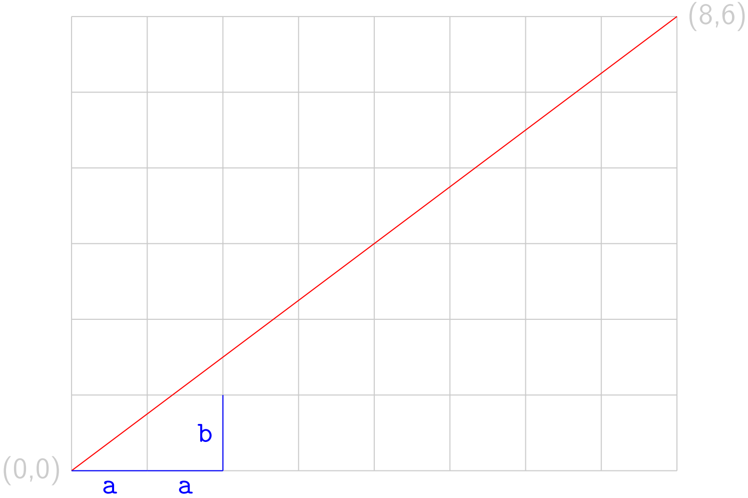
Lyndon words: a 2D point of view



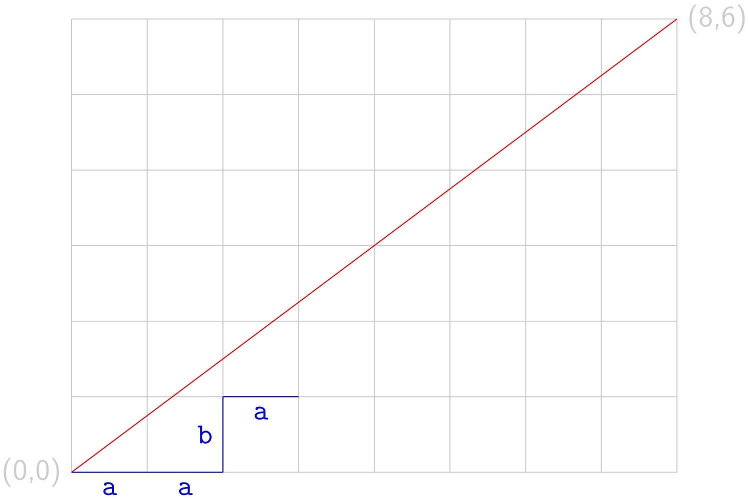
Lyndon words: a 2D point of view



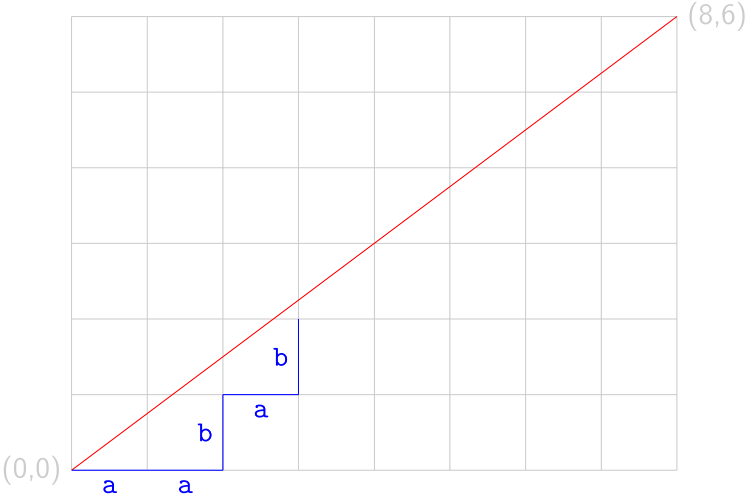
Lyndon words: a 2D point of view



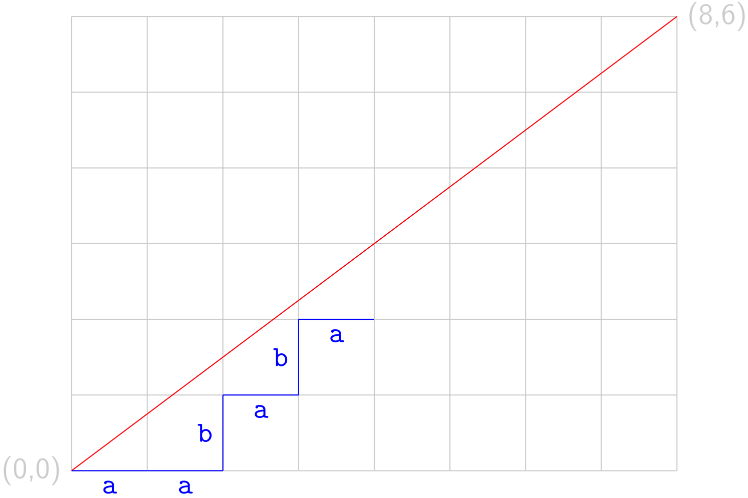
Lyndon words: a 2D point of view



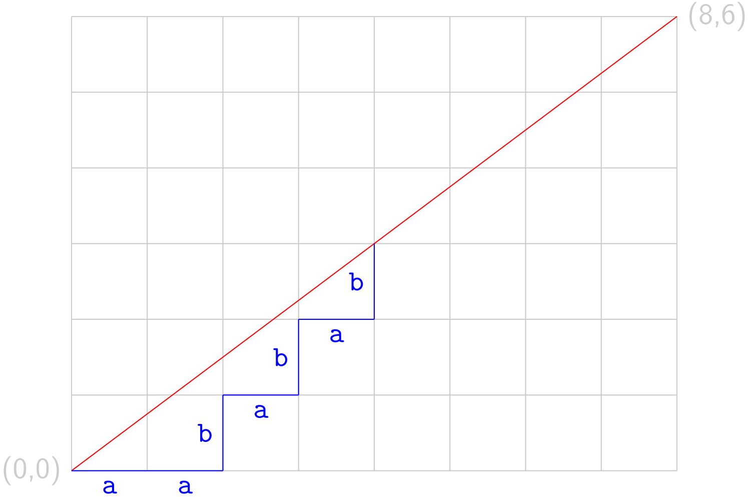
Lyndon words: a 2D point of view



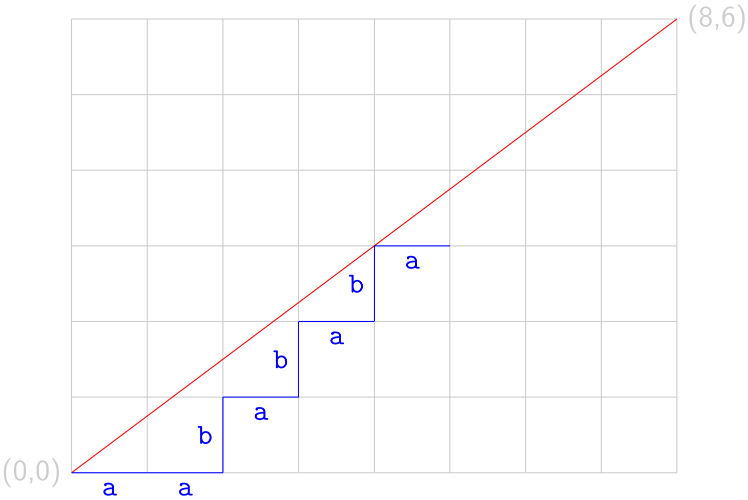
Lyndon words: a 2D point of view



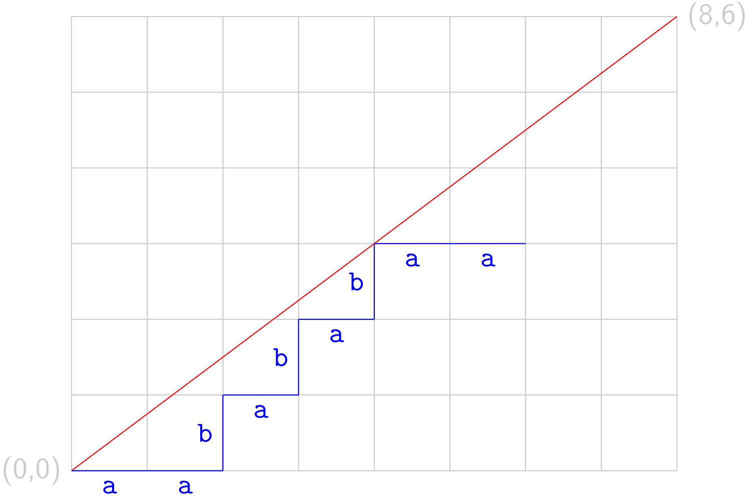
Lyndon words: a 2D point of view



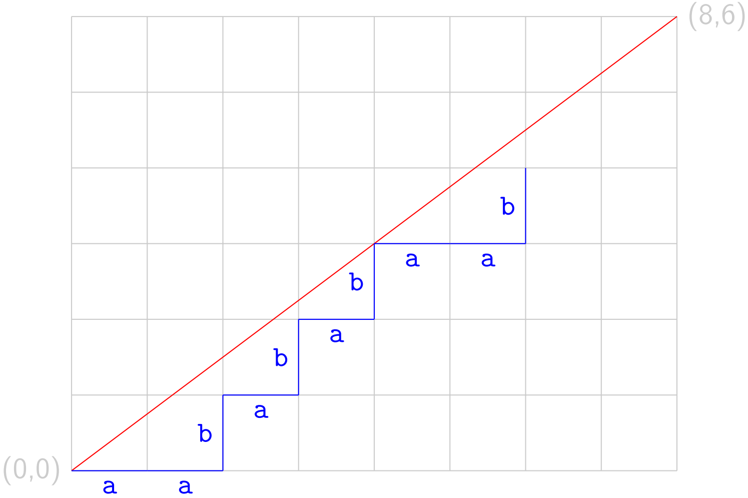
Lyndon words: a 2D point of view



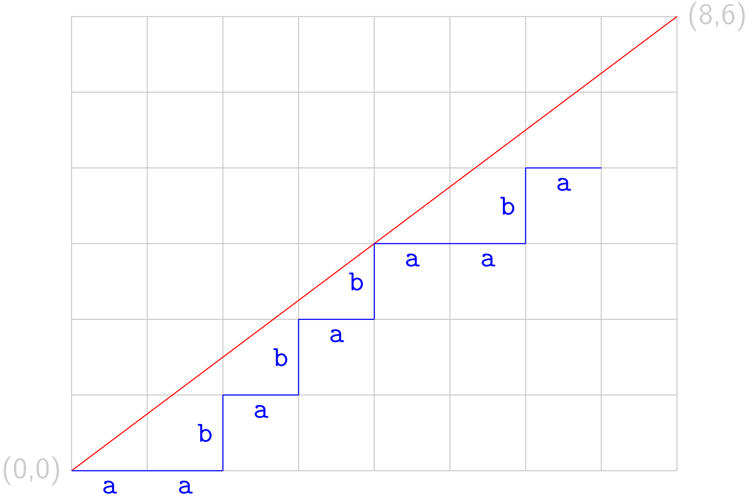
Lyndon words: a 2D point of view



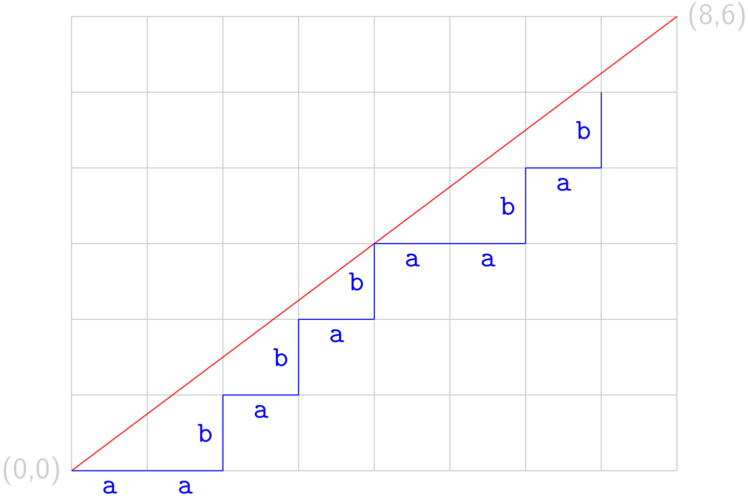
Lyndon words: a 2D point of view



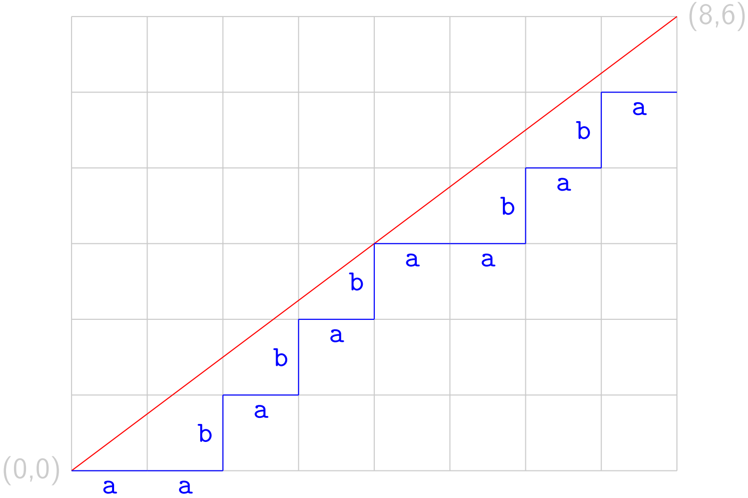
Lyndon words: a 2D point of view



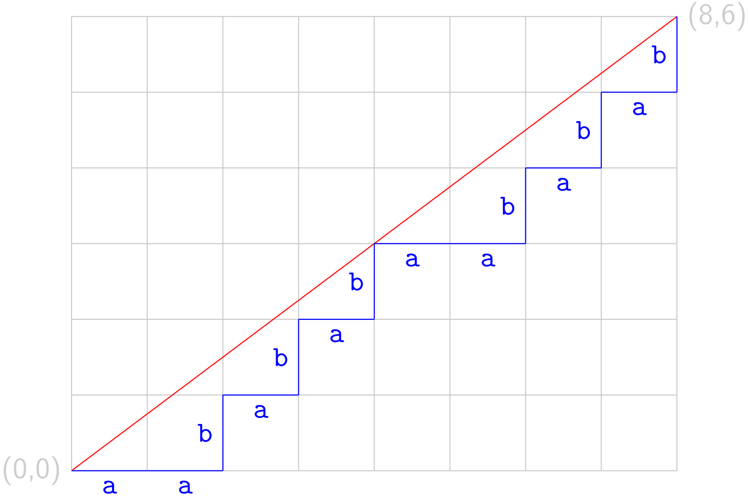
Lyndon words: a 2D point of view



Lyndon words: a 2D point of view

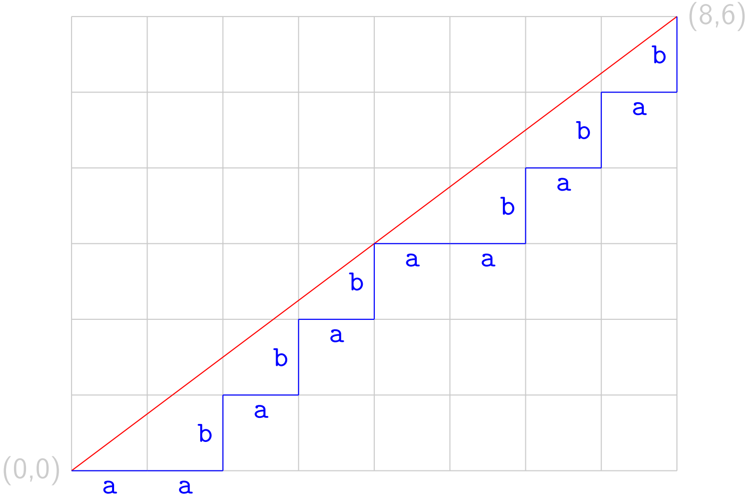


Lyndon words: a 2D point of view

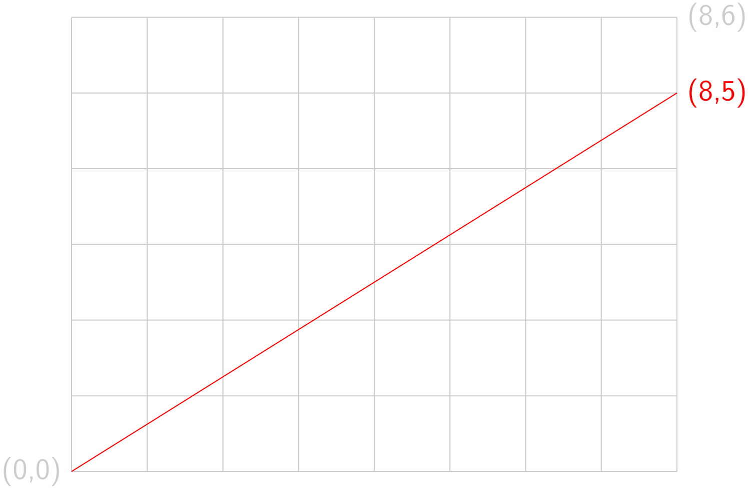


Lyndon words: a 2D point of view

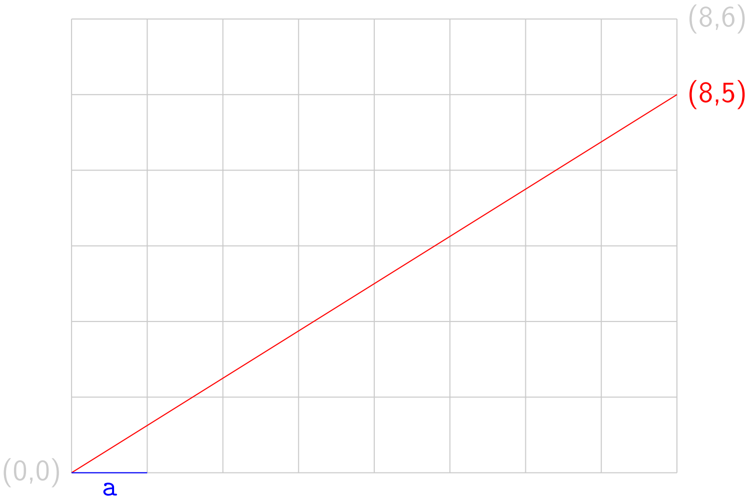
rational slope $\frac{y}{x}$ = Christoffel word



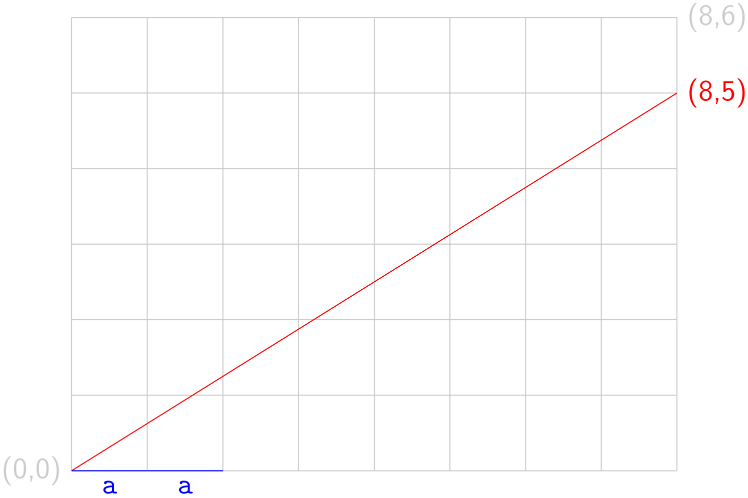
Lyndon words: a 2D point of view



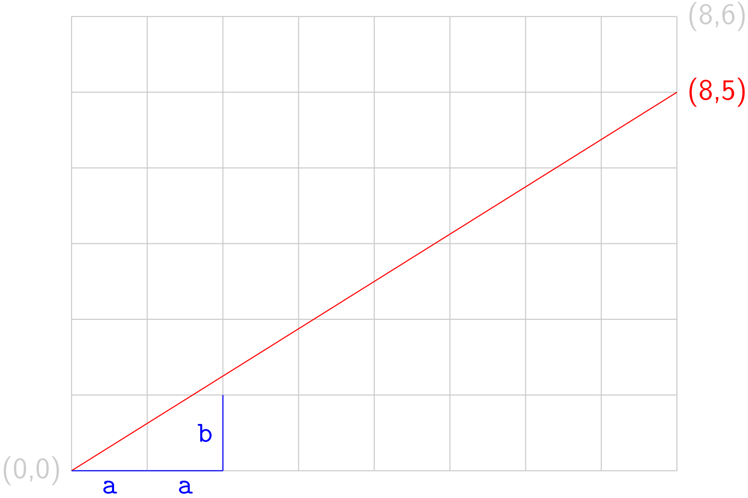
Lyndon words: a 2D point of view



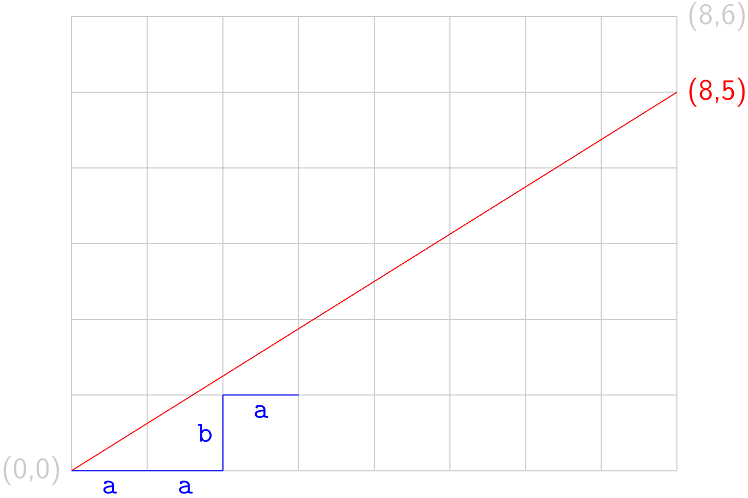
Lyndon words: a 2D point of view



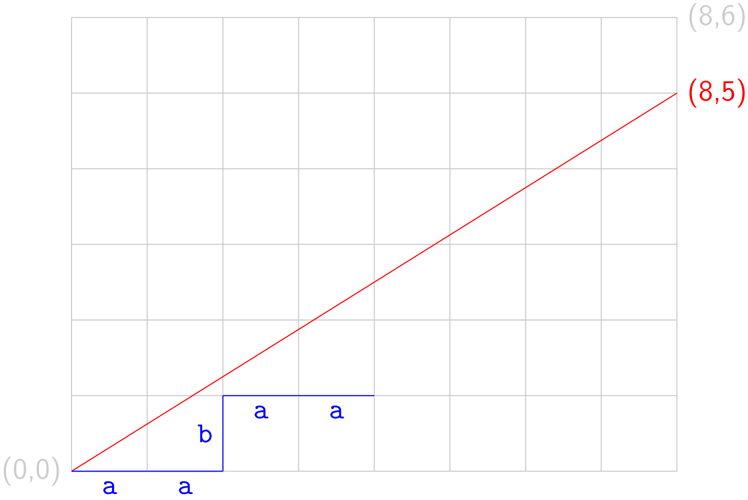
Lyndon words: a 2D point of view



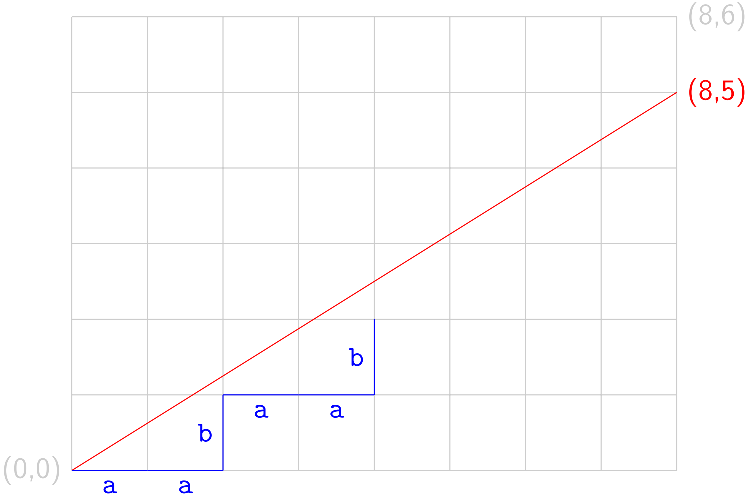
Lyndon words: a 2D point of view



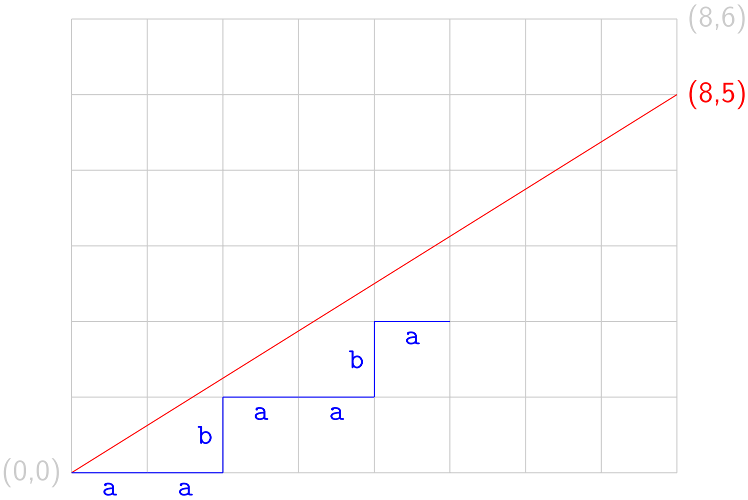
Lyndon words: a 2D point of view



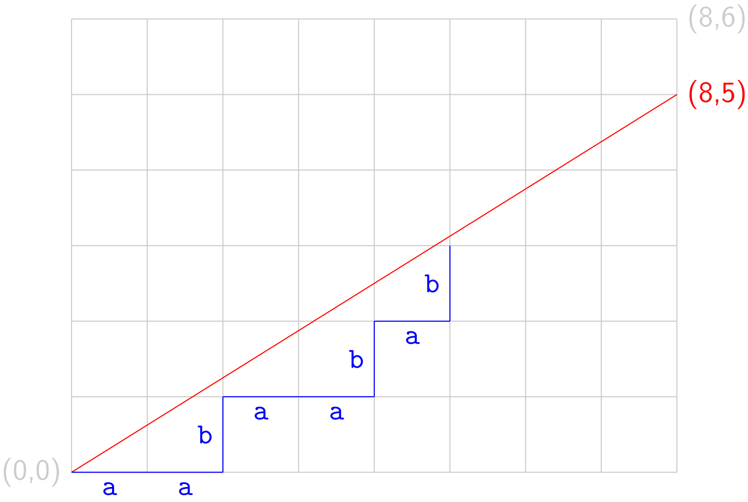
Lyndon words: a 2D point of view



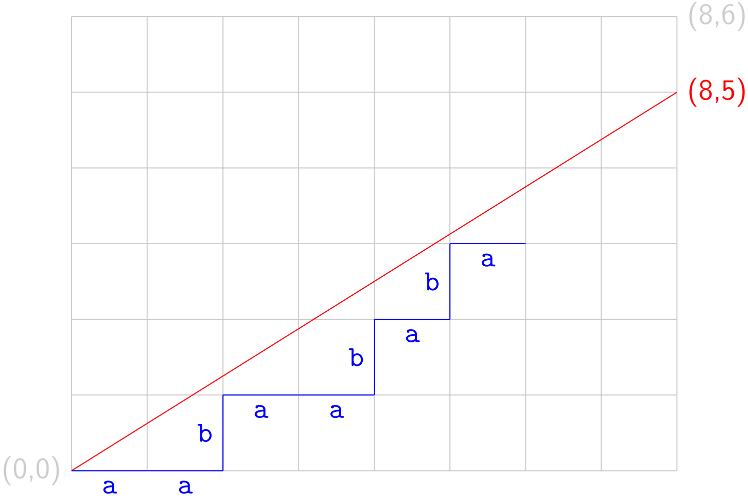
Lyndon words: a 2D point of view



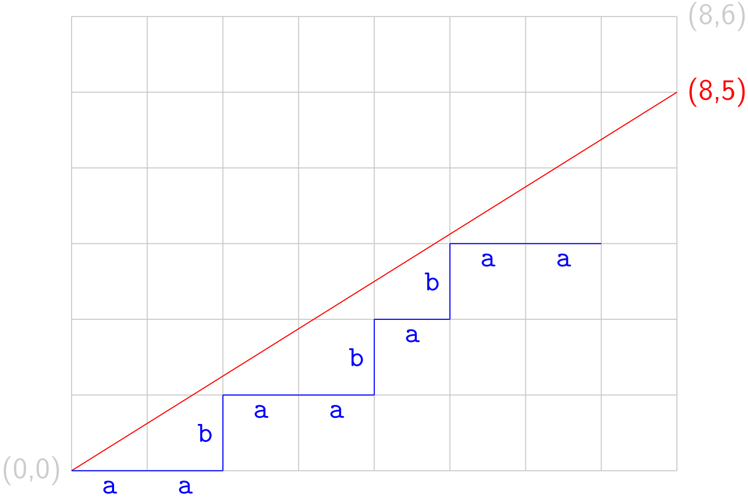
Lyndon words: a 2D point of view



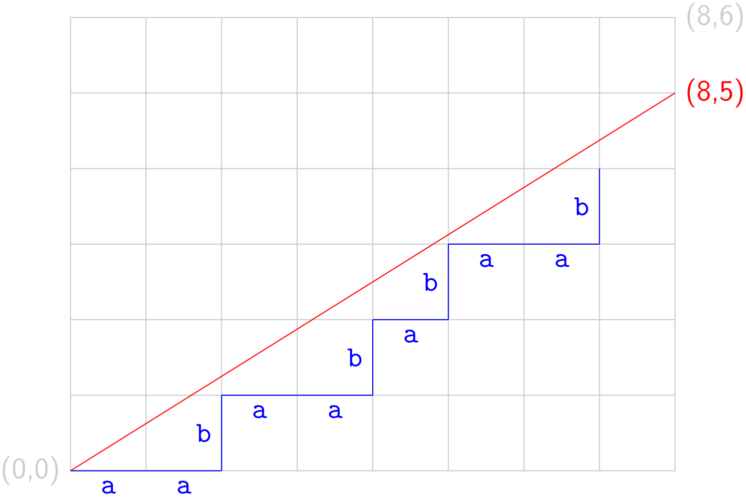
Lyndon words: a 2D point of view



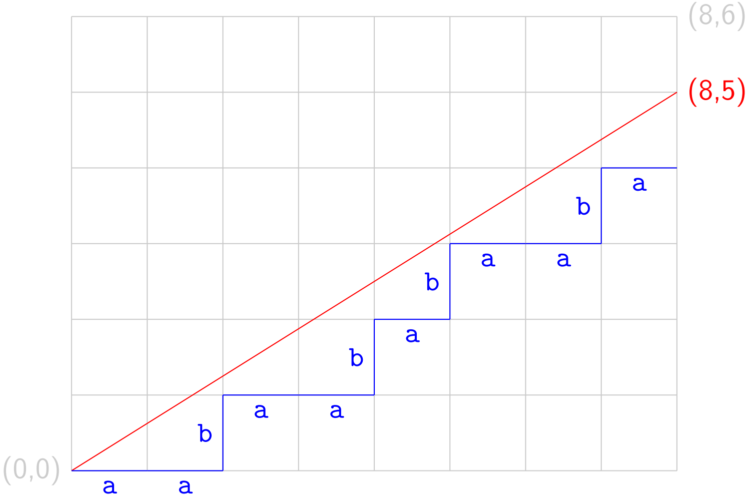
Lyndon words: a 2D point of view



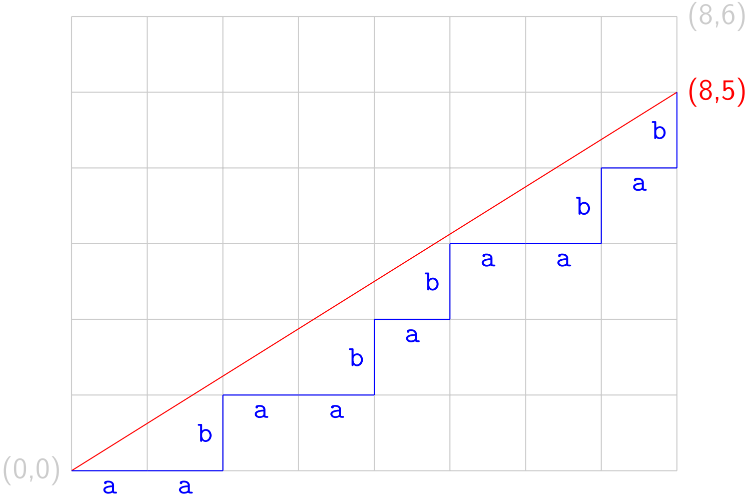
Lyndon words: a 2D point of view



Lyndon words: a 2D point of view

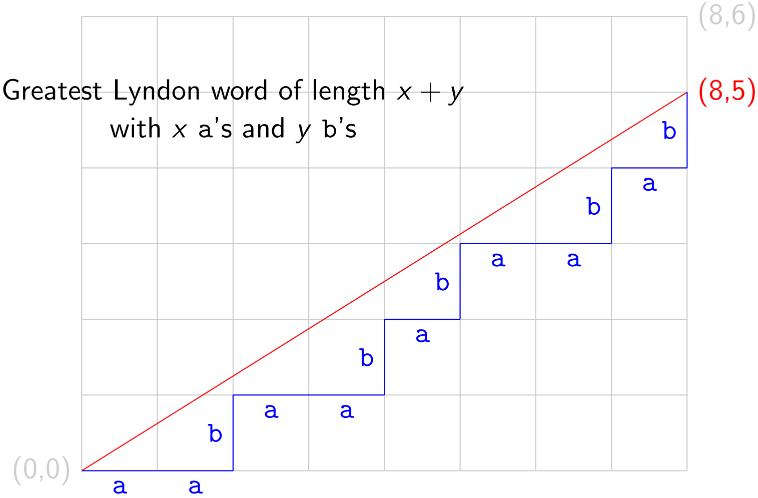


Lyndon words: a 2D point of view



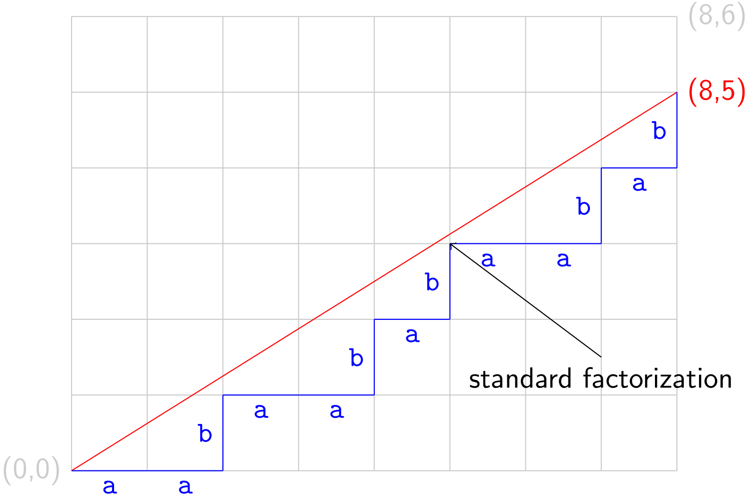
Lyndon words: a 2D point of view

rational slope $\frac{y}{x}$ with x and y co-prime = Lyndon word



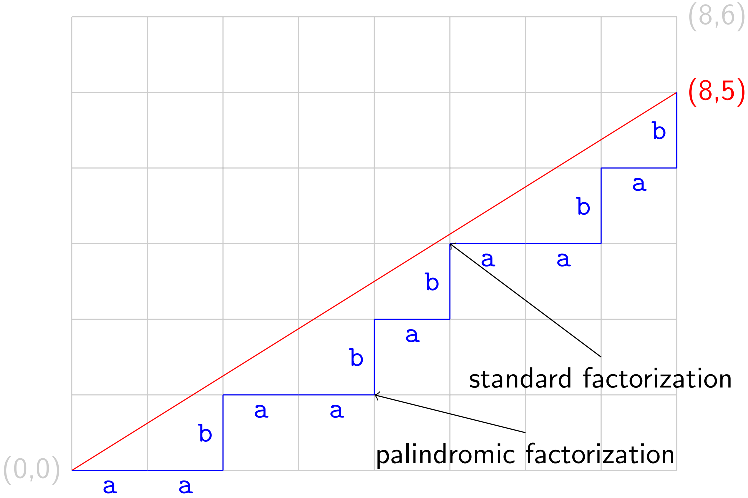
Lyndon words: a 2D point of view

closest point = standard factorization

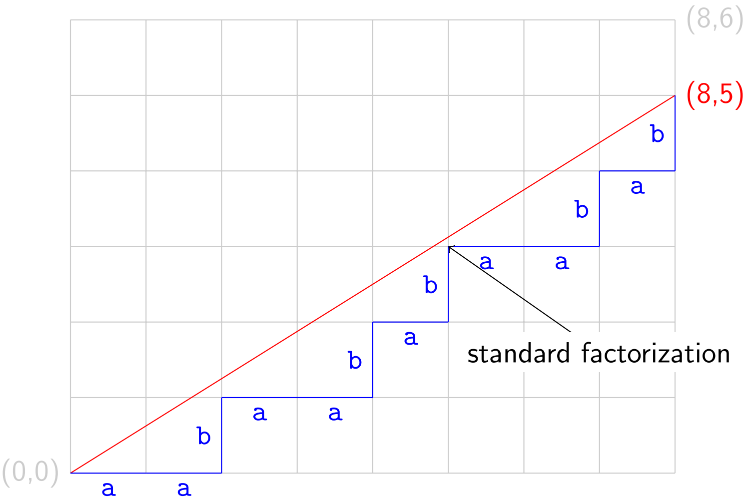


Lyndon words: a 2D point of view

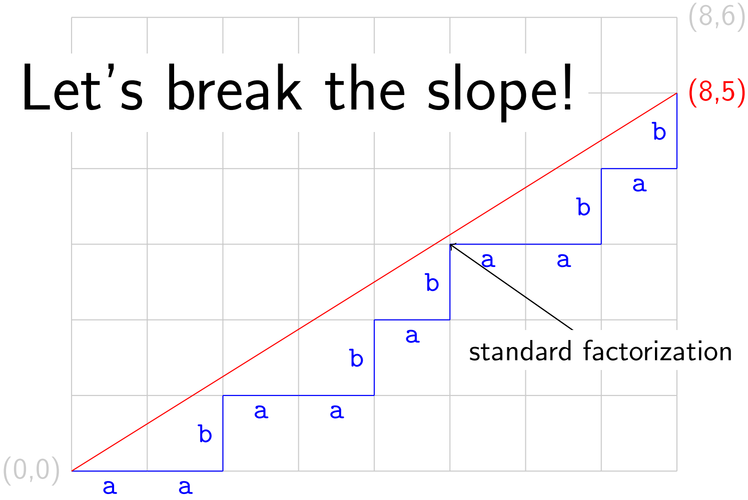
more distant point = palindromic factorization



Lyndon words: a 2D point of view

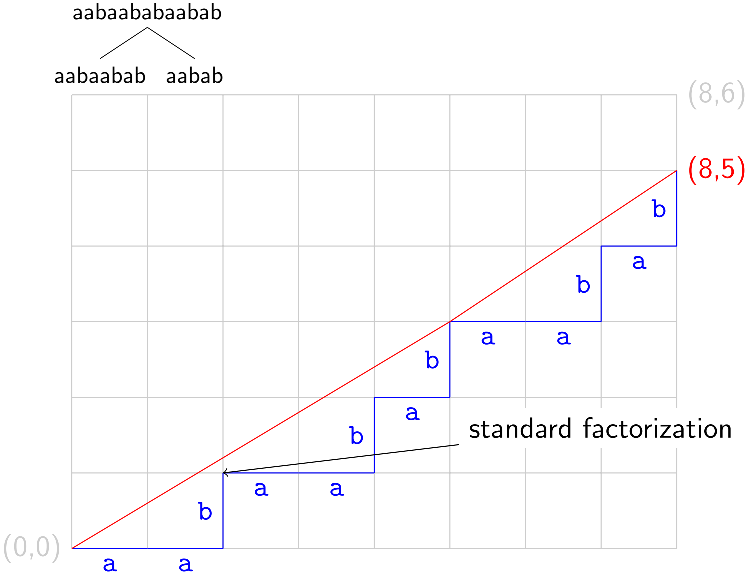


Lyndon words: a 2D point of view

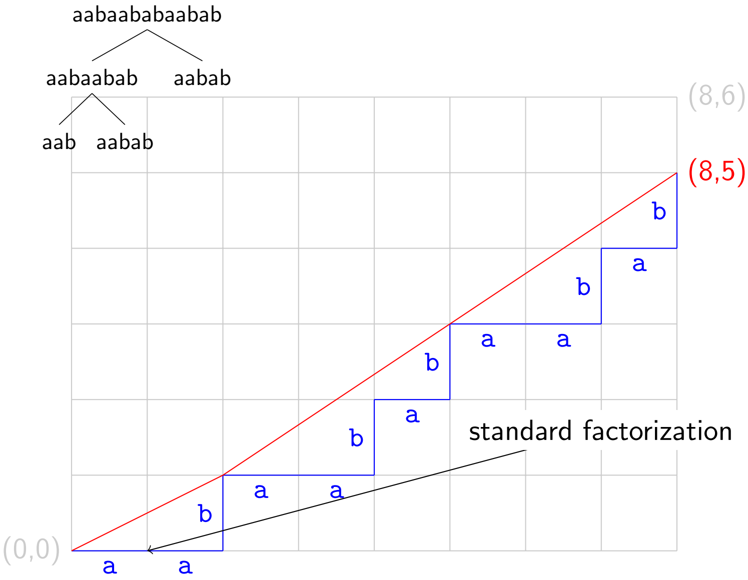


Let's break the slope!

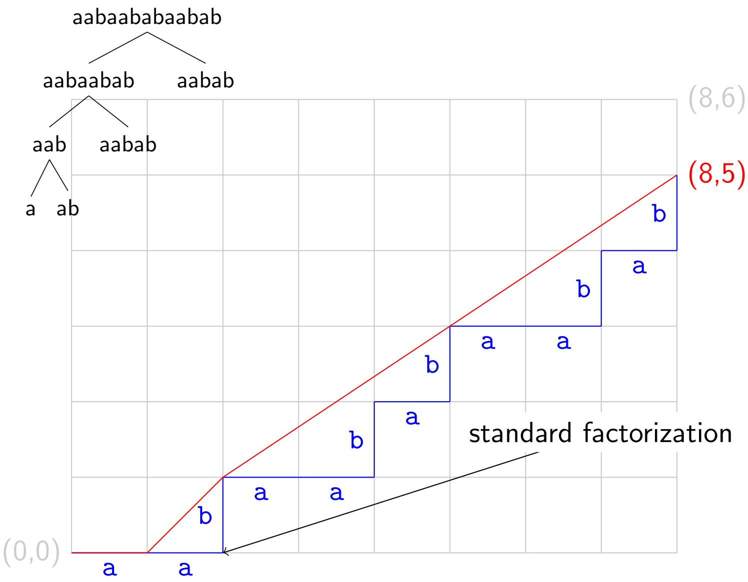
Lyndon words: a 2D point of view



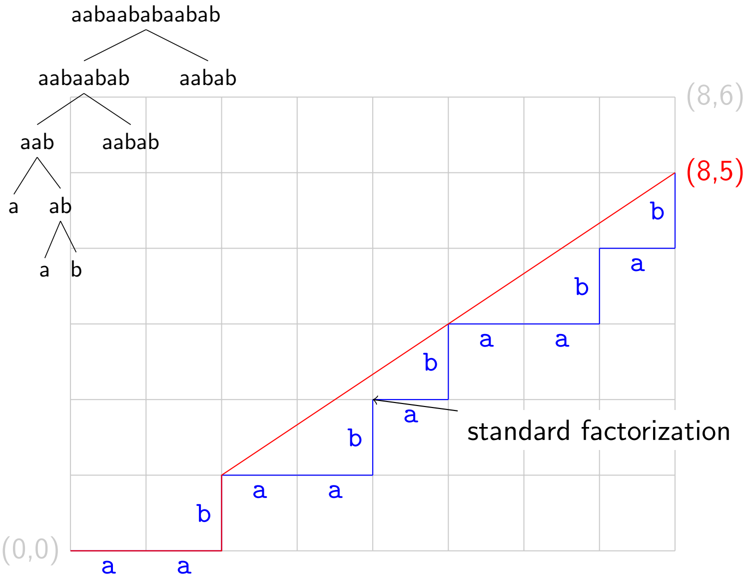
Lyndon words: a 2D point of view



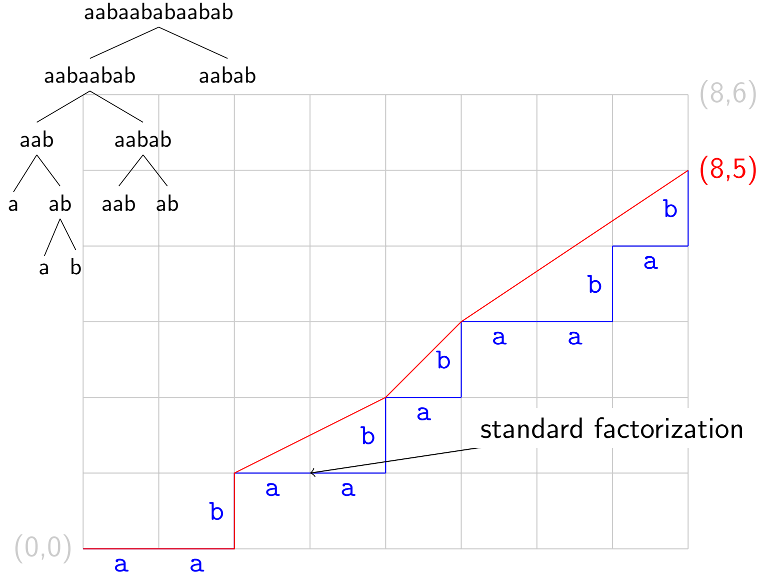
Lyndon words: a 2D point of view



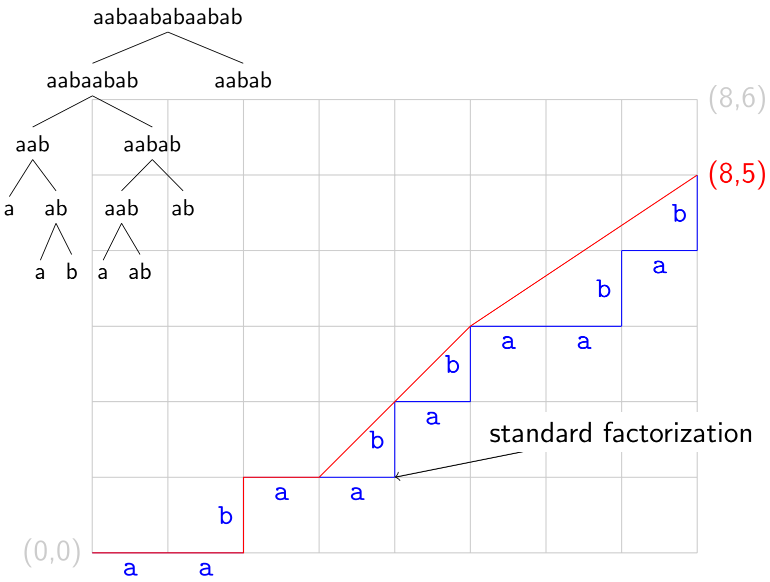
Lyndon words: a 2D point of view



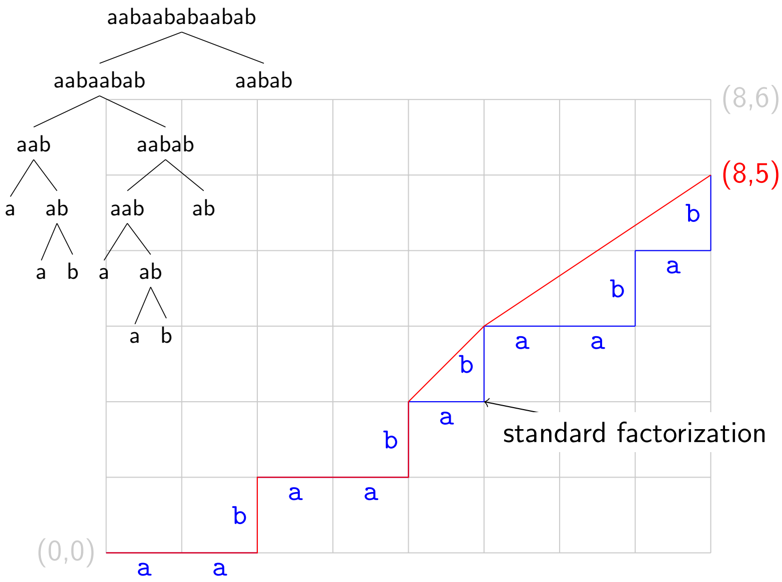
Lyndon words: a 2D point of view



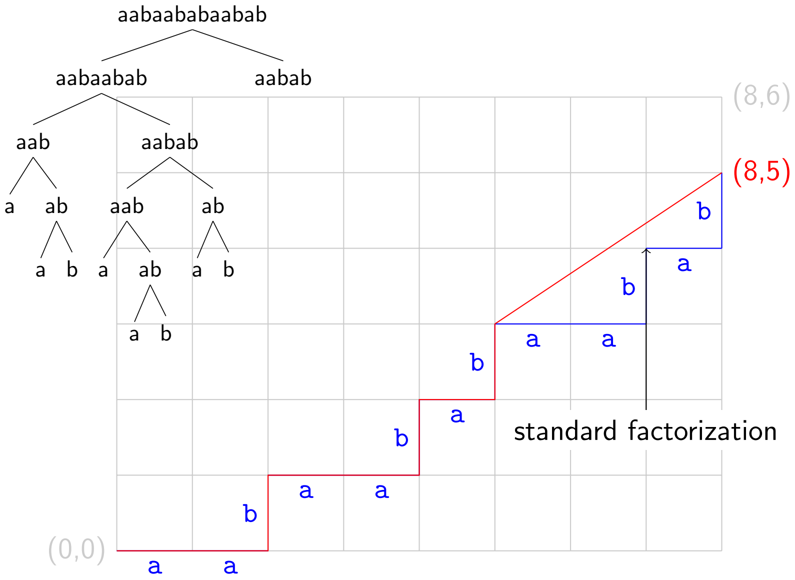
Lyndon words: a 2D point of view



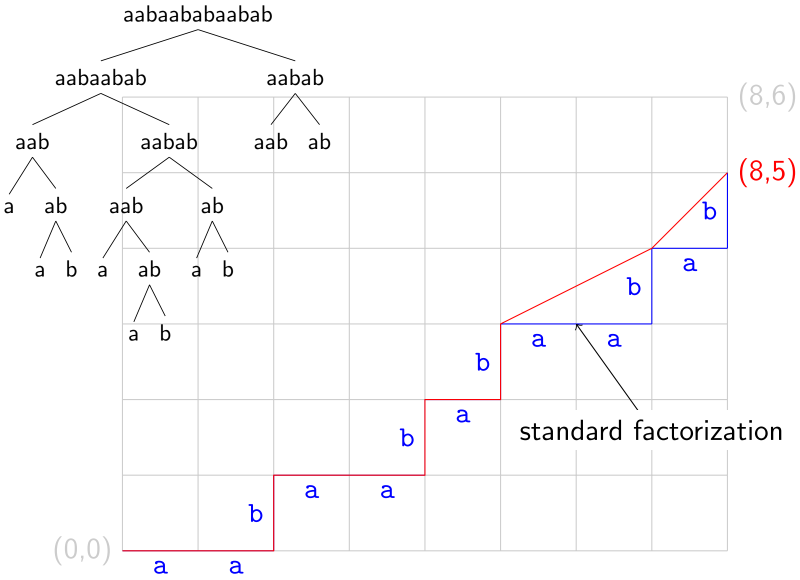
Lyndon words: a 2D point of view



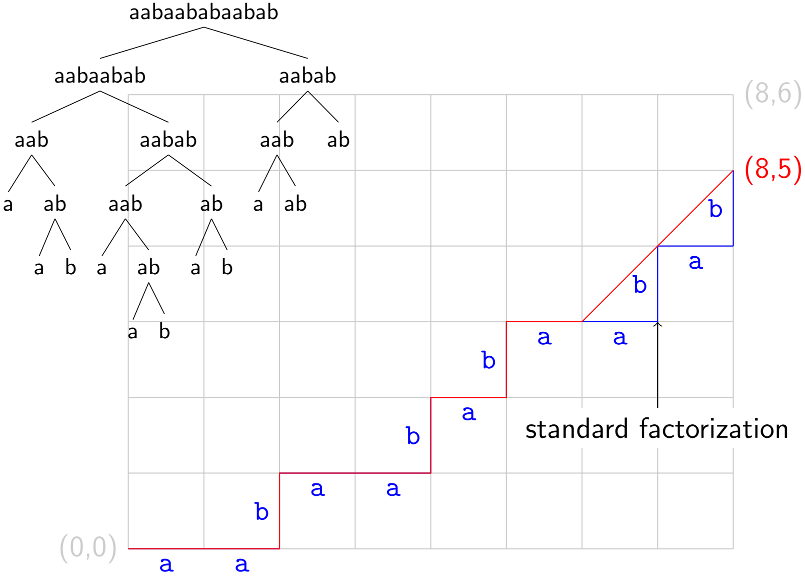
Lyndon words: a 2D point of view



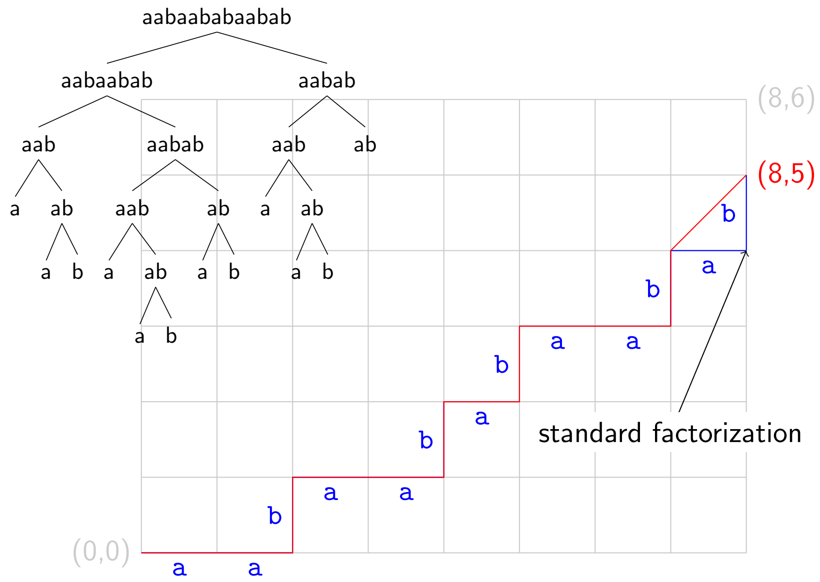
Lyndon words: a 2D point of view



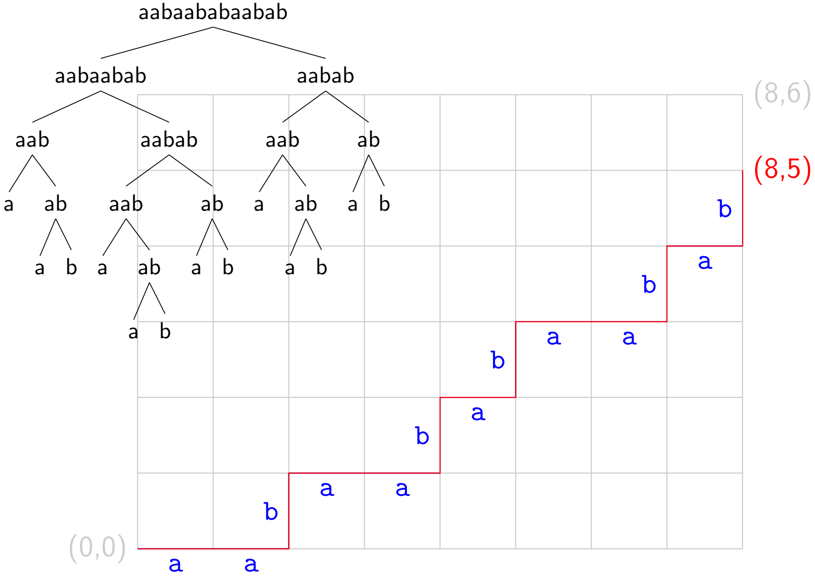
Lyndon words: a 2D point of view



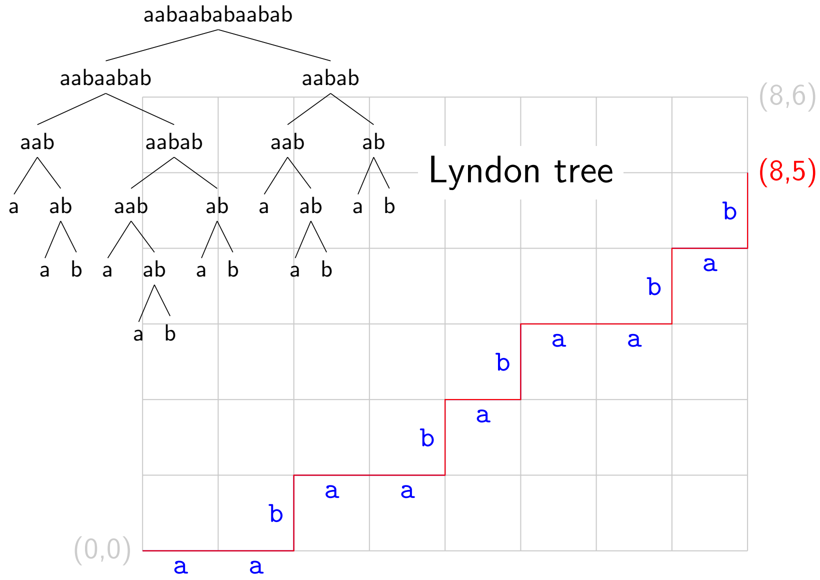
Lyndon words: a 2D point of view



Lyndon words: a 2D point of view

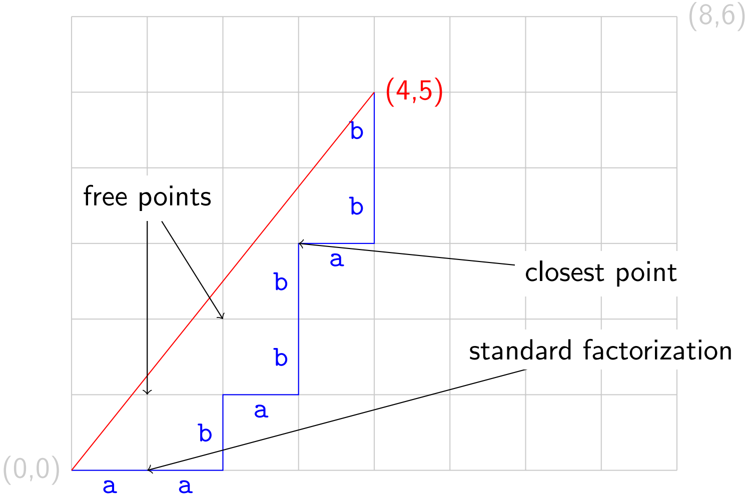


Lyndon words: a 2D point of view

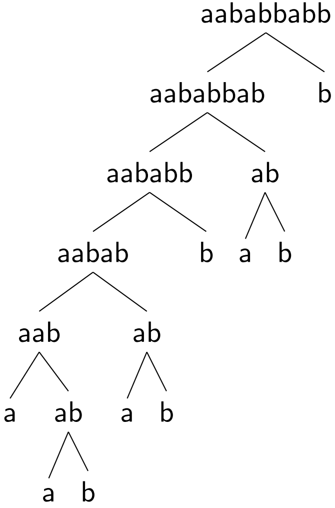


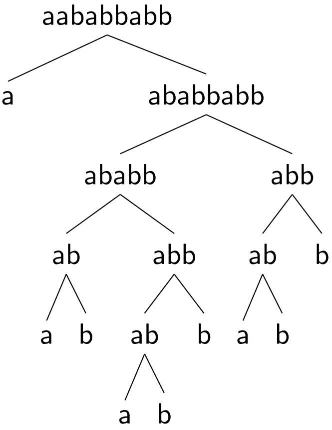
Lyndon words: a 2D point of view

aababbabb is a Lyndon word but...



Lyndon tree and left Lyndon tree





The Chen-Fox-Lyndon Theorem

In 1958, Chen, Fox and Lyndon established that any word `w` can be uniquely factored in a non increasing sequence of Lyndon words:

`w = a b b a b b a b b a b`

The Chen-Fox-Lyndon Theorem

In 1958, Chen, Fox and Lyndon established that any word w can be uniquely factored in a non increasing sequence of Lyndon words:

w = a b b a b b a b b a b

a b b

The Chen-Fox-Lyndon Theorem

In 1958, Chen, Fox and Lyndon established that any word $\mathtt{w}$ can be uniquely factored in a non increasing sequence of Lyndon words:

$$\mathtt{w} = \mathtt{abb}\,\mathtt{abb}\,\mathtt{abbab}$$

$$\mathtt{abb} \geq_{\text{LEX}} \mathtt{abb}$$

The Chen-Fox-Lyndon Theorem

In 1958, Chen, Fox and Lyndon established that any word w can be uniquely factored in a non increasing sequence of Lyndon words:

w = a b b | a b b | a b b | a b

a b b $\geq_{\text{LEX}}$ a b b $\geq_{\text{LEX}}$ a b b

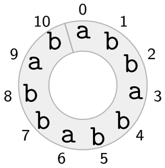
The Chen-Fox-Lyndon Theorem

In 1958, Chen, Fox and Lyndon established that any word $\mathtt{w}$ can be uniquely factored in a non increasing sequence of Lyndon words:

$$\mathtt{w} = \mathtt{abb}|\mathtt{abb}|\mathtt{abb}|\mathtt{ab}$$

$$\mathtt{abb} \geq_{\text{LEX}} \mathtt{abb} \geq_{\text{LEX}} \mathtt{abb} \geq_{\text{LEX}} \mathtt{ab}$$

Booth (1980)



Finds the least circular substring (based on Knuth-Morris-Pratt algorithm, 1977).

Factorization into Lyndon words [Duval,83]

$\mathtt{w} = \mathtt{b\,b\,a\,b\,b\,a\,b\,a\,b\,a\,b\,a\,a\,b\,a\,a\,a}$

Let $\mathtt{w}_i$ and $\mathtt{w}_j$ be two letters at positions $i < j$:
Case 1: $\mathtt{w}_i = \mathtt{w}_j$ then next current factor has a border
Case 2: $\mathtt{w}_i <_{\text{LEX}} \mathtt{w}_j$ then next current factor is a Lyndon word
Case 3: $\mathtt{w}_i >_{\text{LEX}} \mathtt{w}_j$ then current factor can be factored
into Lyndon word(s) (according to its period)

Factorization into Lyndon words [Duval,83]

current factor

$\mathtt{w} = \mathtt{b\,b\,a\,b\,b\,a\,b\,a\,b\,a\,b\,a\,a\,b\,a\,a\,a}$

Let $\mathtt{w}_i$ and $\mathtt{w}_j$ be two letters at positions $i < j$:

→ Case 1: $\mathtt{w}_i = \mathtt{w}_j$ then next current factor has a border

Case 2: $\mathtt{w}_i <_{\text{LEX}} \mathtt{w}_j$ then next current factor is a Lyndon word

Case 3: $\mathtt{w}_i >_{\text{LEX}} \mathtt{w}_j$ then current factor can be factored into Lyndon word(s) (according to its period)

Factorization into Lyndon words [Duval,83]

current factor

w = | b b a b b a b a b a b a a b a a a

Let $\mathtt{w}_i$ and $\mathtt{w}_j$ be two letters at positions $i < j$:

Case 1: $\mathtt{w}_i = \mathtt{w}_j$ then next current factor has a border

Case 2: $\mathtt{w}_i <_{\text{LEX}} \mathtt{w}_j$ then next current factor is a Lyndon word

$\rightarrow$ Case 3: $\mathtt{w}_i >_{\text{LEX}} \mathtt{w}_j$ then current factor can be factored into Lyndon word(s) (according to its period)

Factorization into Lyndon words [Duval,83]

current factor

w = | b | b | a b b a b a b a b a a b a a a

Let $\mathtt{w}_i$ and $\mathtt{w}_j$ be two letters at positions $i < j$:

Case 1: $\mathtt{w}_i = \mathtt{w}_j$ then next current factor has a border

Case 2: $\mathtt{w}_i <_{\text{LEX}} \mathtt{w}_j$ then next current factor is a Lyndon word

Case 3: $\mathtt{w}_i >_{\text{LEX}} \mathtt{w}_j$ then current factor can be factored into Lyndon word(s) (according to its period)

Factorization into Lyndon words [Duval,83]

current factor

$\mathtt{w} = \mathtt{b\,b\,a\,b\,b\,a\,b\,a\,b\,a\,b\,a\,a\,b\,a\,a\,a}$

Let $\mathtt{w}_i$ and $\mathtt{w}_j$ be two letters at positions $i < j$:

Case 1: $\mathtt{w}_i = \mathtt{w}_j$ then next current factor has a border

→ Case 2: $\mathtt{w}_i <_{\text{LEX}} \mathtt{w}_j$ then next current factor is a Lyndon word

Case 3: $\mathtt{w}_i >_{\text{LEX}} \mathtt{w}_j$ then current factor can be factored into Lyndon word(s) (according to its period)

Factorization into Lyndon words [Duval,83]

current factor

$\mathtt{w} = \mathtt{b\,b\,a\,b\,b\,a\,b\,a\,b\,a\,b\,a\,a\,b\,a\,a\,a}$

Let $\mathtt{w}_i$ and $\mathtt{w}_j$ be two letters at positions $i < j$:

Case 1: $\mathtt{w}_i = \mathtt{w}_j$ then next current factor has a border

→ Case 2: $\mathtt{w}_i <_{\text{LEX}} \mathtt{w}_j$ then next current factor is a Lyndon word

Case 3: $\mathtt{w}_i >_{\text{LEX}} \mathtt{w}_j$ then current factor can be factored into Lyndon word(s) (according to its period)

Factorization into Lyndon words [Duval,83]

current factor

$\mathtt{w} = \mathtt{b\,b\,|\,a\,b\,b\,a\,b\,a\,b\,a\,b\,a\,a\,b\,a\,a\,a}$

Let $\mathtt{w}_i$ and $\mathtt{w}_j$ be two letters at positions $i < j$:

→ Case 1: $\mathtt{w}_i = \mathtt{w}_j$ then next current factor has a border

Case 2: $\mathtt{w}_i <_{\text{LEX}} \mathtt{w}_j$ then next current factor is a Lyndon word

Case 3: $\mathtt{w}_i >_{\text{LEX}} \mathtt{w}_j$ then current factor can be factored into Lyndon word(s) (according to its period)

Factorization into Lyndon words [Duval,83]

current factor

$\mathtt{w} = \mathtt{b}\,\mathtt{b}\,|\,\mathtt{a}\,\mathtt{b}\,\mathtt{b}\,\mathtt{a}\,\mathtt{b}\,\mathtt{a}\,\mathtt{b}\,\mathtt{a}\,\mathtt{b}\,\mathtt{a}\,\mathtt{a}\,\mathtt{b}\,\mathtt{a}\,\mathtt{a}\,\mathtt{a}$

Let $\mathtt{w}_i$ and $\mathtt{w}_j$ be two letters at positions $i < j$:

$\rightarrow$ Case 1: $\mathtt{w}_i = \mathtt{w}_j$ then next current factor has a border

Case 2: $\mathtt{w}_i <_{\text{LEX}} \mathtt{w}_j$ then next current factor is a Lyndon word

Case 3: $\mathtt{w}_i >_{\text{LEX}} \mathtt{w}_j$ then current factor can be factored into Lyndon word(s) (according to its period)

Factorization into Lyndon words [Duval,83]

current factor →

$\mathtt{w} =$ b b | a b b a b a b a b a a b a a a

Let $\mathtt{w}_i$ and $\mathtt{w}_j$ be two letters at positions $i < j$:

Case 1: $\mathtt{w}_i = \mathtt{w}_j$ then next current factor has a border

Case 2: $\mathtt{w}_i <_{\text{LEX}} \mathtt{w}_j$ then next current factor is a Lyndon word

→ Case 3: $\mathtt{w}_i >_{\text{LEX}} \mathtt{w}_j$ then current factor can be factored into Lyndon word(s) (according to its period)

Factorization into Lyndon words [Duval,83]

current factor

w = b b a b b a b a b a b a a b a a b a a a

Let w_i and w_j be two letters at positions $i < j$:

Case 1: $w_i = w_j$ then next current factor has a border

Case 2: $w_i <_{LEX} w_j$ then next current factor is a Lyndon word

→ Case 3: $w_i >_{LEX} w_j$ then current factor can be factored into Lyndon word(s) (according to its period)

Factorization into Lyndon words [Duval,83]

current factor

$\mathtt{w} = \mathtt{b\,b\,a\,b\,b\,|\,a\,b\,a\,b\,a\,b\,a\,a\,b\,a\,a\,a}$

Let $\mathtt{w}_i$ and $\mathtt{w}_j$ be two letters at positions $i < j$:

Case 1: $\mathtt{w}_i = \mathtt{w}_j$ then next current factor has a border

→ Case 2: $\mathtt{w}_i <_{\text{LEX}} \mathtt{w}_j$ then next current factor is a Lyndon word

Case 3: $\mathtt{w}_i >_{\text{LEX}} \mathtt{w}_j$ then current factor can be factored into Lyndon word(s) (according to its period)

Factorization into Lyndon words [Duval,83]

current factor

$\mathtt{w} = \mathtt{b\,b\,a\,b\,b\,a\,b\,a\,b\,a\,b\,a\,a\,b\,a\,a\,a}$

Let $\mathtt{w}_i$ and $\mathtt{w}_j$ be two letters at positions $i < j$:

$\rightarrow$ Case 1: $\mathtt{w}_i = \mathtt{w}_j$ then next current factor has a border

Case 2: $\mathtt{w}_i <_{\text{LEX}} \mathtt{w}_j$ then next current factor is a Lyndon word

Case 3: $\mathtt{w}_i >_{\text{LEX}} \mathtt{w}_j$ then current factor can be factored into Lyndon word(s) (according to its period)

Factorization into Lyndon words [Duval,83]

current factor

w = b b a b b | a b a b a b a a b a a a

Let $\mathtt{w}_i$ and $\mathtt{w}_j$ be two letters at positions $i < j$:

→ Case 1: $\mathtt{w}_i = \mathtt{w}_j$ then next current factor has a border

Case 2: $\mathtt{w}_i <_{\text{LEX}} \mathtt{w}_j$ then next current factor is a Lyndon word

Case 3: $\mathtt{w}_i >_{\text{LEX}} \mathtt{w}_j$ then current factor can be factored into Lyndon word(s) (according to its period)

Factorization into Lyndon words [Duval,83]

current factor

$\mathtt{w}$ = b b a b b | a b a b a b a a b a a a

Let $\mathtt{w}_i$ and $\mathtt{w}_j$ be two letters at positions $i < j$:

→ Case 1: $\mathtt{w}_i = \mathtt{w}_j$ then next current factor has a border

Case 2: $\mathtt{w}_i <_{\text{LEX}} \mathtt{w}_j$ then next current factor is a Lyndon word

Case 3: $\mathtt{w}_i >_{\text{LEX}} \mathtt{w}_j$ then current factor can be factored into Lyndon word(s) (according to its period)

Factorization into Lyndon words [Duval,83]

current factor

$w = $ b b a b b | a b a b a b a a b a a a

Let w_i and w_j be two letters at positions $i < j$:

→ Case 1: $w_i = w_j$ then next current factor has a border

Case 2: $w_i <_{\text{LEX}} w_j$ then next current factor is a Lyndon word

Case 3: $w_i >_{\text{LEX}} w_j$ then current factor can be factored into Lyndon word(s) (according to its period)

Factorization into Lyndon words [Duval,83]

current factor

$w = $ b b a b b | a b a b a b a a b a a a

Let w_i and w_j be two letters at positions $i < j$:

→ Case 1: $w_i = w_j$ then next current factor has a border

Case 2: $w_i <_{\text{LEX}} w_j$ then next current factor is a Lyndon word

Case 3: $w_i >_{\text{LEX}} w_j$ then current factor can be factored into Lyndon word(s) (according to its period)

Factorization into Lyndon words [Duval,83]

current factor

w = b b a b b | a b a b a b a a b a a a

Let w_i and w_j be two letters at positions $i < j$:

Case 1: $w_i = w_j$ then next current factor has a border

Case 2: $w_i <_{\text{LEX}} w_j$ then next current factor is a Lyndon word

→ Case 3: $w_i >_{\text{LEX}} w_j$ then current factor can be factored into Lyndon word(s) (according to its period)

Factorization into Lyndon words [Duval,83]

current factor

$w = $ b b a b b | a b a b a b a a b a a a

Let w_i and w_j be two letters at positions $i < j$:

Case 1: $w_i = w_j$ then next current factor has a border

Case 2: $w_i <_{\text{LEX}} w_j$ then next current factor is a Lyndon word

→ Case 3: $w_i >_{\text{LEX}} w_j$ then current factor can be factored into Lyndon word(s) (according to its period)

Factorization into Lyndon words [Duval,83]

current factor

$\mathtt{w} = \mathtt{b\,b\,a\,b\,b\,a\,b\,a\,b\,a\,b\,a\,a\,b\,a\,a\,a}$

Let $\mathtt{w}_i$ and $\mathtt{w}_j$ be two letters at positions $i < j$:

→ Case 1: $\mathtt{w}_i = \mathtt{w}_j$ then next current factor has a border

Case 2: $\mathtt{w}_i <_{\text{LEX}} \mathtt{w}_j$ then next current factor is a Lyndon word

Case 3: $\mathtt{w}_i >_{\text{LEX}} \mathtt{w}_j$ then current factor can be factored into Lyndon word(s) (according to its period)

Factorization into Lyndon words [Duval,83]

current factor

$\mathtt{w}$ = b b a b b a b a b a b a a b a a a

Let $\mathtt{w}_i$ and $\mathtt{w}_j$ be two letters at positions $i < j$:

Case 1: $\mathtt{w}_i = \mathtt{w}_j$ then next current factor has a border

→ Case 2: $\mathtt{w}_i <_{\text{LEX}} \mathtt{w}_j$ then next current factor is a Lyndon word

Case 3: $\mathtt{w}_i >_{\text{LEX}} \mathtt{w}_j$ then current factor can be factored into Lyndon word(s) (according to its period)

Factorization into Lyndon words [Duval,83]

current factor

$\mathtt{w} = \mathtt{b\,b\,a\,b\,b\,a\,b\,a\,b\,a\,b\,|\,a\,a\,b\,a\,a\,a}$

Let $\mathtt{w}_i$ and $\mathtt{w}_j$ be two letters at positions $i < j$:

$\rightarrow$ Case 1: $\mathtt{w}_i = \mathtt{w}_j$ then next current factor has a border

Case 2: $\mathtt{w}_i <_{\text{LEX}} \mathtt{w}_j$ then next current factor is a Lyndon word

Case 3: $\mathtt{w}_i >_{\text{LEX}} \mathtt{w}_j$ then current factor can be factored into Lyndon word(s) (according to its period)

Factorization into Lyndon words [Duval,83]

current factor

w = b b a b b a b a b a b | a a b a a a

Let $\mathtt{w}_i$ and $\mathtt{w}_j$ be two letters at positions $i < j$:

→ Case 1: $\mathtt{w}_i = \mathtt{w}_j$ then next current factor has a border

Case 2: $\mathtt{w}_i <_{\text{LEX}} \mathtt{w}_j$ then next current factor is a Lyndon word

Case 3: $\mathtt{w}_i >_{\text{LEX}} \mathtt{w}_j$ then current factor can be factored into Lyndon word(s) (according to its period)

Factorization into Lyndon words [Duval,83]

current factor

$w = $ b b a b b a b a b a b | a a b a a a

Let w_i and w_j be two letters at positions $i < j$:

Case 1: $w_i = w_j$ then next current factor has a border

Case 2: $w_i <_{\text{LEX}} w_j$ then next current factor is a Lyndon word

$\rightarrow$ Case 3: $w_i >_{\text{LEX}} w_j$ then current factor can be factored into Lyndon word(s) (according to its period)

Factorization into Lyndon words [Duval,83]

current factor

$\mathtt{w} = $ b b a b b a b a b a b | a a b a a a

Let $\mathtt{w}_i$ and $\mathtt{w}_j$ be two letters at positions $i < j$:

Case 1: $\mathtt{w}_i = \mathtt{w}_j$ then next current factor has a border

Case 2: $\mathtt{w}_i <_{\text{LEX}} \mathtt{w}_j$ then next current factor is a Lyndon word

$\rightarrow$ Case 3: $\mathtt{w}_i >_{\text{LEX}} \mathtt{w}_j$ then current factor can be factored into Lyndon word(s) (according to its period)

Factorization into Lyndon words [Duval,83]

current factor

w = b b a b b a b a b a b a a b a a a

Let w_i and w_j be two letters at positions $i < j$:

$\rightarrow$ Case 1: $w_i = w_j$ then next current factor has a border

Case 2: $w_i <_{\text{LEX}} w_j$ then next current factor is a Lyndon word

Case 3: $w_i >_{\text{LEX}} w_j$ then current factor can be factored into Lyndon word(s) (according to its period)

Factorization into Lyndon words [Duval,83]

current factor

w = b b a b b a b a b a b a a b | a a a

Let $\mathtt{w}_i$ and $\mathtt{w}_j$ be two letters at positions $i < j$:

→ Case 1: $\mathtt{w}_i = \mathtt{w}_j$ then next current factor has a border

Case 2: $\mathtt{w}_i <_{\text{LEX}} \mathtt{w}_j$ then next current factor is a Lyndon word

Case 3: $\mathtt{w}_i >_{\text{LEX}} \mathtt{w}_j$ then current factor can be factored into Lyndon word(s) (according to its period)

Factorization into Lyndon words [Duval,83]

current factor

$\mathtt{w} = \mathtt{bbabbababababaab|aaa}$

Let $\mathtt{w}_i$ and $\mathtt{w}_j$ be two letters at positions $i < j$:

Case 1: $\mathtt{w}_i = \mathtt{w}_j$ then next current factor has a border

Case 2: $\mathtt{w}_i <_{\text{LEX}} \mathtt{w}_j$ then next current factor is a Lyndon word

Case 3: $\mathtt{w}_i >_{\text{LEX}} \mathtt{w}_j$ then current factor can be factored into Lyndon word(s) (according to its period)

Factorization into Lyndon words [Duval,83]

current factor

$\mathtt{w} = $ b b a b b a b a b a b a a b | a a a

Let $\mathtt{w}_i$ and $\mathtt{w}_j$ be two letters at positions $i < j$:

Case 1: $\mathtt{w}_i = \mathtt{w}_j$ then next current factor has a border

Case 2: $\mathtt{w}_i <_{\text{LEX}} \mathtt{w}_j$ then next current factor is a Lyndon word

Case 3: $\mathtt{w}_i >_{\text{LEX}} \mathtt{w}_j$ then current factor can be factored into Lyndon word(s) (according to its period)

Factorization into Lyndon words [Duval,83]

w = | b | b | a | b | b | a | b | a | b | a | b | a | a | b | a | a | a |

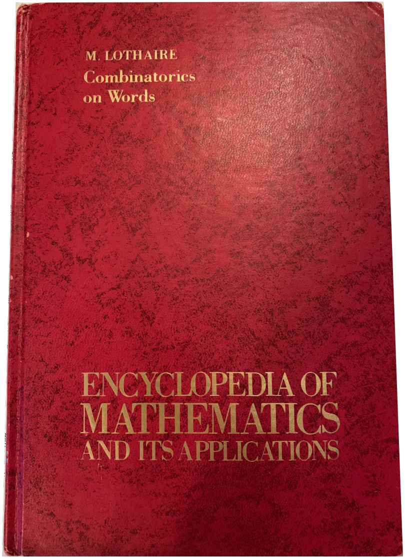
Factorization is performed:

- online
- in linear time
- with constant extra space (3 integers)

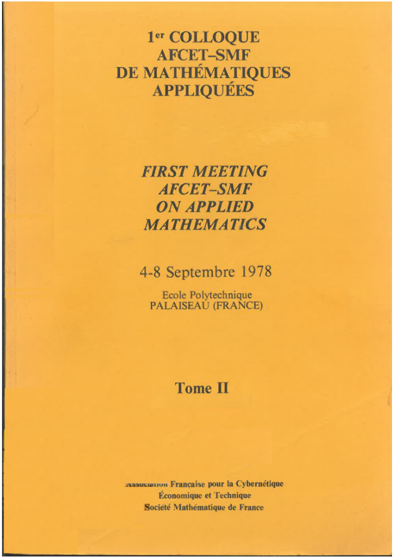
But. . .

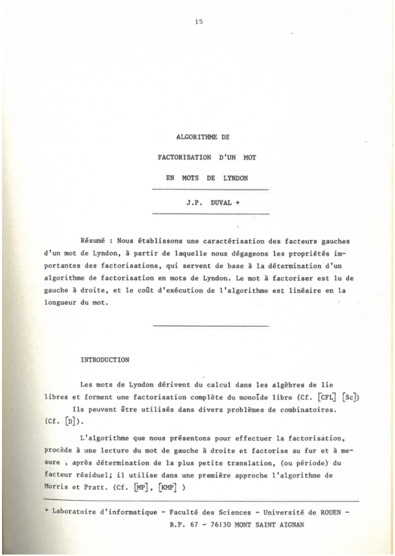
. . . where do the Lyndon words come from?

Lothaire, 1982



The origin of "Lyndon words"





15

ALGORITHME DE

FACTORISATION D'UN MOT

EN MOTS DE LYNDON

J.P. DUVAL +

Résumé : Nous établissons une caractérisation des facteurs gauches d'un mot de Lyndon, à partir de laquelle nous dégageons les propriétés importantes des factorisations, qui servent de base à la détermination d'un algorithme de factorisation en mots de Lyndon. Le mot à factoriser est lu de gauche à droite, et le coût d'exécution de l'algorithme est linéaire en la longueur du mot.

INTRODUCTION

Les mots de Lyndon dérivent du calcul dans les algèbres de lie libres et forment une factorisation complète du monoïde libre (Cf. [CFL] [Sc])

Ils peuvent être utilisés dans divers problèmes de combinatoires. (Cf. [D]).

L'algorithme que nous présentons pour effectuer la factorisation, procède à une lecture du mot de gauche à droite et factorise au fur et à mesure , après détermination de la plus petite translation, (ou période) du facteur résiduel; il utilise dans une première approche l'algorithme de Morris et Pratt. (Cf. [MP], [KMP])

+ Laboratoire d'informatique - Faculté des Sciences - Université de ROUEN - B.P. 67 - 76130 MONT SAINT AIGNAN

Suffix permutation → Lyndon factorization

	↓	↓	↓			↓		↓		↓		↓			↓	↓	↓
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
w =	b	b	a	b	b	a	b	a	b	a	b	a	a	b	a	a	a
sp(w) =	16	14	8	15	13	7	12	6	11	5	10	3	4	9	2	1	0

Factorization $\rightarrow$ suffix permutation [Mantaci et al,2013]

	↓	↓	↓			↓		↓		↓		↓			↓	↓	↓
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
w =	b	b	a	b	b	a	b	a	b	a	b	a	a	b	a	a	a
sp(w) =	16	14	8	15	13	7	12	6	11	5	10	3	4	9	2	1	0

Given the Lyndon factorization of a word, the relative order of two suffixes inside one of these factors is the same as their relative order in the whole word.

Burrows-Wheeler Transform Scottified [Scott,2007]

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
w =	b	b	a	b	b	a	b	a	b	a	b	a	a	b	a	a	a

Burrows-Wheeler Transform Scottified [Scott,2007]

w =

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
b	b	a	b	b	a	b	a	b	a	b	a	a	b	a	a	a

Burrows-Wheeler Transform Scottified [Scott,2007]

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
w =	b	b	a	b	b	a	b	a	b	a	b	a	a	b	a	a	a

b
b

a b b

a b

a b

a b

a a b

a
a
a

Burrows-Wheeler Transform Scottified [Scott,2007]

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
w =	b	b	a	b	b	a	b	a	b	a	b	a	a	b	a	a	a

b   b

b

a b b

a b

a b   conjugates $\longrightarrow$

a b

a a b

a

a

a

Burrows-Wheeler Transform Scottified [Scott,2007]

w =

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
b	b	a	b	b	a	b	a	b	a	b	a	a	b	a	a	a

b b

b b

a b b

a b

a b $\xrightarrow{\text{conjugates}}$

a b

a a b

a

a

a

Burrows-Wheeler Transform Scottified [Scott,2007]

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
w =	b	b	a	b	b	a	b	a	b	a	b	a	a	b	a	a	a

b
b
a b b
a b
a b
a b
a a b
a
a
a

conjugates →

b
b
a b b
b a b
b b a

Burrows-Wheeler Transform Scottified [Scott,2007]

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
w =	b	b	a	b	b	a	b	a	b	a	b	a	a	b	a	a	a

b
b
a b b
a b
a b
a b
a a b
a
a
a

conjugates →

b
b
a b b
b a b
b b a
a b
b a

Burrows-Wheeler Transform Scottified [Scott,2007]

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
w =	b	b	a	b	b	a	b	a	b	a	b	a	a	b	a	a	a

b		b
b		b
		a b b
a b b		b a b
		b b a
a b		a b
		b a
a b	conjugates →	a b
		b a
a b		
a a b		
a		
a		
a		

Burrows-Wheeler Transform Scottified [Scott,2007]

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
w =	b	b	a	b	b	a	b	a	b	a	b	a	a	b	a	a	a

Lyndon factors		Conjugates
b		b
b		b
		a b b
a b b		b a b
		b b a
a b		a b
		b a
a b	conjugates →	a b
		b a
a b		a b
		b a
a a b		
a		
a		
a		

Burrows-Wheeler Transform Scottified [Scott,2007]

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
w =	b	b	a	b	b	a	b	a	b	a	b	a	a	b	a	a	a

b
b
a b b
a b
a b
a b
a a b
a
a
a

conjugates ⟶

b
b
a b b
b a b
b b a
a b
b a
a b
b a
a b
b a
a a b
a b a
b a a

Burrows-Wheeler Transform Scottified [Scott,2007]

w =

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
b	b	a	b	b	a	b	a	b	a	b	a	a	b	a	a	a

b		b
b		b
		a b b
a b b		b a b
		b b a
a b		a b
		b a
a b	conjugates ⟶	a b
		b a
a b		a b
		b a
		a a b
a a b		a b a
		b a a
a		a
a		
a		

Burrows-Wheeler Transform Scottified [Scott,2007]

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
w =	b	b	a	b	b	a	b	a	b	a	b	a	a	b	a	a	a

b
b
a b b
a b
a b
a b
a a b
a
a
a

conjugates ⟶

b
b
a b b
b a b
b b a
a b
b a
a b
b a
a b
b a
a a b
a b a
b a a
a
a

Burrows-Wheeler Transform Scottified [Scott,2007]

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
w =	b	b	a	b	b	a	b	a	b	a	b	a	a	b	a	a	a

Factors		Conjugates
b		b
b		b
a b b		a b b
		b a b
		b b a
a b		a b
		b a
a b	conjugates ⟶	a b
		b a
a b		a b
		b a
a a b		a a b
		a b a
		b a a
a		a
a		a
a		a

Burrows-Wheeler Transform Scottified [Scott,2007]

w =

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
b	b	a	b	b	a	b	a	b	a	b	a	a	b	a	a	a

b
b
a b b
a b
a b
a b
a a b
a
a
a

conjugates →

b
b
a b b
b a b
b b a
a b
b a
a b
b a
a b
b a
a a b
a b a
b a a
a
a
a

sort →

a
a
a
a a b
a b a
a b
a b
a b
a b b
b a a
b a
b a
b a
b a b
b b a
b
b

Burrows-Wheeler Transform Scottified [Scott,2007]

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
w =	b	b	a	b	b	a	b	a	b	a	b	a	a	b	a	a	a

	conjugates →	sort →	bwts →
b	b	a	a
b	b	a	a
a b b	a b b	a	a
	b a b	a a b	b
	b b a	a b a	a
a b	a b	a b	b
	b a	a b	b
a b	a b	a b	b
	b a	a b b	b
a b	a b	b a a	a
	b a	b a	a
a a b	a a b	b a	a
	a b a	b a	a
	b a a	b a b	b
a	a	b b a	a
a	a	b	b
a	a	b	b

Burrows-Wheeler Transform Scottified [Scott,2007]

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
w =	b	b	a	b	b	a	b	a	b	a	b	a	a	b	a	a	a

0	a	a	0
1	a	a	1
2	a	a	2
3	a	b	9
4	a	a	3
5	a	b	10
6	a	b	11
7	a	b	12
8	a	b	13
9	b	a	4
10	b	a	5
11	b	a	6
12	b	a	7
13	b	b	15
14	b	a	8
15	b	b	14
16	b	b	16

Burrows-Wheeler Transform Scottified [Scott,2007]

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
w =	b	b	a	b	b	a	b	a	b	a	b	a	a	b	a	a	a

0	a	⟷	a	0	(0)
1	a		a	1	
2	a		a	2	
3	a		b	9	
4	a		a	3	
5	a		b	10	
6	a		b	11	
7	a		b	12	
8	a		b	13	
9	b		a	4	
10	b		a	5	
11	b		a	6	
12	b		a	7	
13	b		b	15	
14	b		a	8	
15	b		b	14	
16	b		b	16	

Burrows-Wheeler Transform Scottified [Scott,2007]

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
w =	b	b	a	b	b	a	b	a	b	a	b	a	a	b	a	a	a

0	a	a	0	(0)
1	a	a	1	(1)
2	a	a	2	
3	a	b	9	
4	a	a	3	
5	a	b	10	
6	a	b	11	
7	a	b	12	
8	a	b	13	
9	b	a	4	
10	b	a	5	
11	b	a	6	
12	b	a	7	
13	b	b	15	
14	b	a	8	
15	b	b	14	
16	b	b	16	

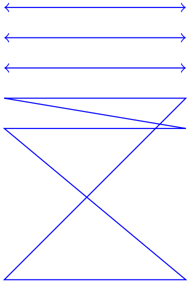
Burrows-Wheeler Transform Scottified [Scott,2007]

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
w =	b	b	a	b	b	a	b	a	b	a	b	a	a	b	a	a	a

0	a	⟷	a	0	(0)
1	a	⟷	a	1	(1)
2	a	⟷	a	2	(2)
3	a		b	9	
4	a		a	3	
5	a		b	10	
6	a		b	11	
7	a		b	12	
8	a		b	13	
9	b		a	4	
10	b		a	5	
11	b		a	6	
12	b		a	7	
13	b		b	15	
14	b		a	8	
15	b		b	14	
16	b		b	16	

Burrows-Wheeler Transform Scottified [Scott,2007]

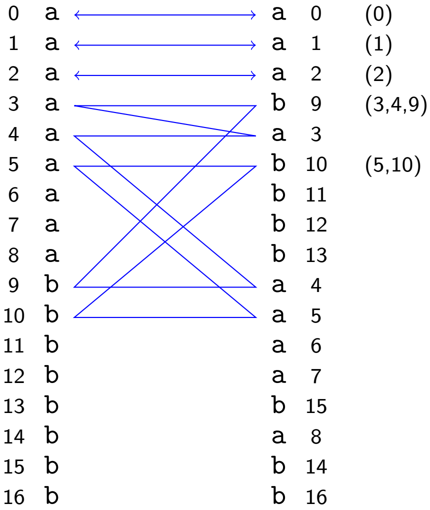
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
w =	b	b	a	b	b	a	b	a	b	a	b	a	a	b	a	a	a



0	a	a	0	(0)
1	a	a	1	(1)
2	a	a	2	(2)
3	a	b	9	(3,4,9)
4	a	a	3	
5	a	b	10	
6	a	b	11	
7	a	b	12	
8	a	b	13	
9	b	a	4	
10	b	a	5	
11	b	a	6	
12	b	a	7	
13	b	b	15	
14	b	a	8	
15	b	b	14	
16	b	b	16	

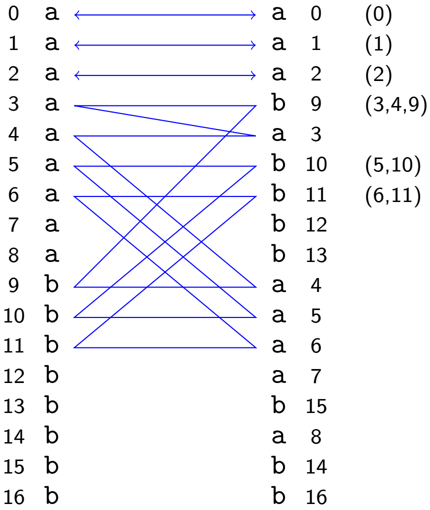
Burrows-Wheeler Transform Scottified [Scott,2007]

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
w =	b	b	a	b	b	a	b	a	b	a	b	a	a	b	a	a	a



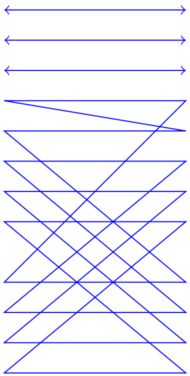
Burrows-Wheeler Transform Scottified [Scott,2007]

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
w =	b	b	a	b	b	a	b	a	b	a	b	a	a	b	a	a	a



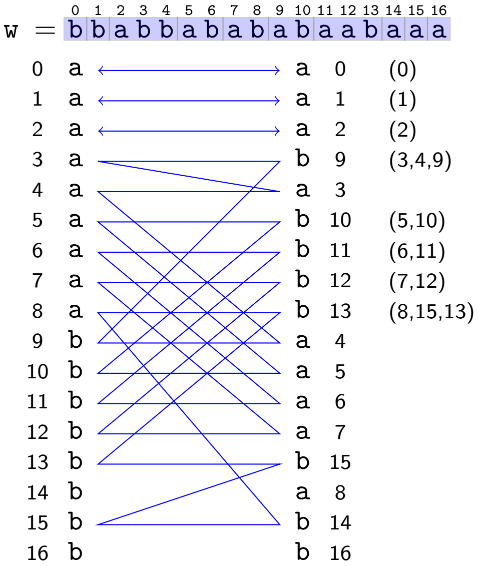
Burrows-Wheeler Transform Scottified [Scott,2007]

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
w =	b	b	a	b	b	a	b	a	b	a	b	a	a	b	a	a	a



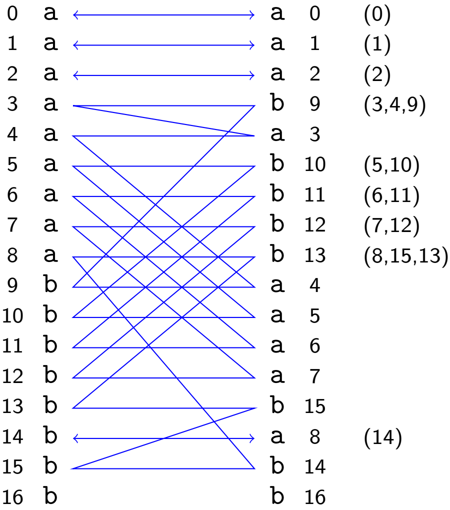
0	a	a	0	(0)
1	a	a	1	(1)
2	a	a	2	(2)
3	a	b	9	(3,4,9)
4	a	a	3	
5	a	b	10	(5,10)
6	a	b	11	(6,11)
7	a	b	12	(7,12)
8	a	b	13	
9	b	a	4	
10	b	a	5	
11	b	a	6	
12	b	a	7	
13	b	b	15	
14	b	a	8	
15	b	b	14	
16	b	b	16	

Burrows-Wheeler Transform Scottified [Scott,2007]



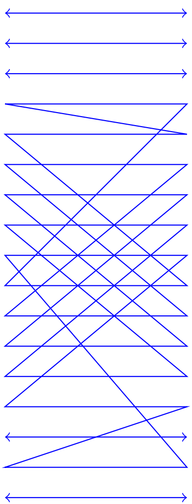
Burrows-Wheeler Transform Scottified [Scott,2007]

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
w =	b	b	a	b	b	a	b	a	b	a	b	a	a	b	a	a	a



Burrows-Wheeler Transform Scottified [Scott,2007]

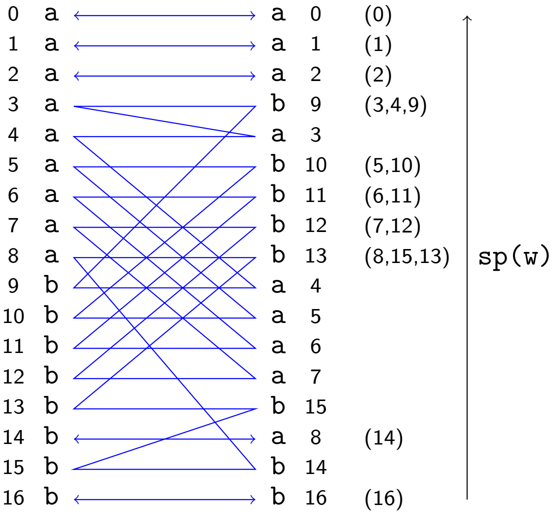
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
w =	b	b	a	b	b	a	b	a	b	a	b	a	a	b	a	a	a



0	a	a	0	(0)
1	a	a	1	(1)
2	a	a	2	(2)
3	a	b	9	(3,4,9)
4	a	a	3	
5	a	b	10	(5,10)
6	a	b	11	(6,11)
7	a	b	12	(7,12)
8	a	b	13	(8,15,13)
9	b	a	4	
10	b	a	5	
11	b	a	6	
12	b	a	7	
13	b	b	15	
14	b	a	8	(14)
15	b	b	14	
16	b	b	16	(16)

Burrows-Wheeler Transform Scottified [Scott,2007]

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
w =	b	b	a	b	b	a	b	a	b	a	b	a	a	b	a	a	a



Lyndon words generation [Duval,1988]

Lyndon words of length up to 4:

Lyndon words generation [Duval,1988]

Lyndon words of length up to 4:

a

Lyndon words generation [Duval,1988]

Lyndon words of length up to 4:

```
a
a a a a
```

Lyndon words generation [Duval,1988]

Lyndon words of length up to 4:

```
a
a a a b
```

Lyndon words generation [Duval,1988]

Lyndon words of length up to 4:

```
a
a a a b
a a b
```

Lyndon words generation [Duval,1988]

Lyndon words of length up to 4:

```
a
a a a b
a a b
a a b a
```

Lyndon words generation [Duval,1988]

Lyndon words of length up to 4:

```
a
a a a b
a a b
a a b b
```

Lyndon words generation [Duval,1988]

Lyndon words of length up to 4:

a
a a a b
a a b
a a b b
a b

Lyndon words generation [Duval,1988]

Lyndon words of length up to 4:

```
a
a a a b
a a b
a a b b
a b
a b a b
```

Lyndon words generation [Duval,1988]

Lyndon words of length up to 4:

```
a
a a a b
a a b
a a b b
a b
a b b
```

Lyndon words generation [Duval,1988]

Lyndon words of length up to 4:

```
a
a a a b
a a b
a a b b
a b
a b b
a b b a
```

Lyndon words generation [Duval,1988]

Lyndon words of length up to 4:

```
a
a a a b
a a b
a a b b
a b
a b b
a b b b
```

Lyndon words generation [Duval,1988]

Lyndon words of length up to 4:

```
a
a a a b
a a b
a a b b
a b
a b b
a b b b
b
```

de Bruijn word of order n

A de Bruijn word of order n is a circular word containing exactly one occurrence of all possible words of length n.

de Bruijn word of order n

A de Bruijn word of order n is a circular word containing exactly one occurrence of all possible words of length n.
For instance, the words of length 4 over a binary alphabet are:

de Bruijn word of order n

A de Bruijn word of order n is a circular word containing exactly one occurrence of all possible words of length n.
For instance, the words of length 4 over a binary alphabet are:
`a a a a`

de Bruijn word of order n

A de Bruijn word of order n is a circular word containing exactly one occurrence of all possible words of length n.
For instance, the words of length 4 over a binary alphabet are:

`a a a a`
`a a a b`

de Bruijn word of order n

A de Bruijn word of order n is a circular word containing exactly one occurrence of all possible words of length n.
For instance, the words of length 4 over a binary alphabet are:

```
a a a a
a a a b
a a b a
```

de Bruijn word of order n

A de Bruijn word of order n is a circular word containing exactly one occurrence of all possible words of length n.

For instance, the words of length 4 over a binary alphabet are:

```
a a a a
a a a b
a a b a
a a b b
```

de Bruijn word of order n

A de Bruijn word of order n is a circular word containing exactly one occurrence of all possible words of length n.

For instance, the words of length 4 over a binary alphabet are:

```
a a a a
a a a b
a a b a
a a b b
a b a a
```

de Bruijn word of order n

A de Bruijn word of order n is a circular word containing exactly one occurrence of all possible words of length n.

For instance, the words of length 4 over a binary alphabet are:

```
a a a a
a a a b
a a b a
a a b b
a b a a
a b a b
```

de Bruijn word of order n

A de Bruijn word of order n is a circular word containing exactly one occurrence of all possible words of length n.
For instance, the words of length 4 over a binary alphabet are:

```
a a a a
a a a b
a a b a
a a b b
a b a a
a b a b
a b b a
```

de Bruijn word of order n

A de Bruijn word of order n is a circular word containing exactly one occurrence of all possible words of length n.
For instance, the words of length 4 over a binary alphabet are:

```
a a a a
a a a b
a a b a
a a b b
a b a a
a b a b
a b b a
a b b b
```

de Bruijn word of order n

A de Bruijn word of order n is a circular word containing exactly one occurrence of all possible words of length n.

For instance, the words of length 4 over a binary alphabet are:

```
a a a a
a a a b
a a b a
a a b b
a b a a
a b a b
a b b a
a b b b
b a a a
```

de Bruijn word of order n

A de Bruijn word of order n is a circular word containing exactly one occurrence of all possible words of length n.
For instance, the words of length 4 over a binary alphabet are:

```
a a a a
a a a b
a a b a
a a b b
a b a a
a b a b
a b b a
a b b b
b a a a
b a a b
```

de Bruijn word of order *n*

A de Bruijn word of order *n* is a circular word containing exactly one occurrence of all possible words of length *n*.
For instance, the words of length 4 over a binary alphabet are:

```
a a a a
a a a b
a a b a
a a b b
a b a a
a b a b
a b b a
a b b b
b a a a
b a a b
b a b a
```

de Bruijn word of order n

A de Bruijn word of order n is a circular word containing exactly one occurrence of all possible words of length n.
For instance, the words of length 4 over a binary alphabet are:

```
a a a a
a a a b
a a b a
a a b b
a b a a
a b a b
a b b a
a b b b
b a a a
b a a b
b a b a
b a b b
```

de Bruijn word of order n

A de Bruijn word of order n is a circular word containing exactly one occurrence of all possible words of length n.

For instance, the words of length 4 over a binary alphabet are:

```
a a a a
a a a b
a a b a
a a b b
a b a a
a b a b
a b b a
a b b b
b a a a
b a a b
b a b a
b a b b
b b a a
```

de Bruijn word of order n

A de Bruijn word of order n is a circular word containing exactly one occurrence of all possible words of length n.
For instance, the words of length 4 over a binary alphabet are:

```
a a a a
a a a b
a a b a
a a b b
a b a a
a b a b
a b b a
a b b b
b a a a
b a a b
b a b a
b a b b
b b a a
b b a b
```

de Bruijn word of order n

A de Bruijn word of order n is a circular word containing exactly one occurrence of all possible words of length n.
For instance, the words of length 4 over a binary alphabet are:

```
a a a a
a a a b
a a b a
a a b b
a b a a
a b a b
a b b a
a b b b
b a a a
b a a b
b a b a
b a b b
b b a a
b b a b
b b b a
```

de Bruijn word of order n

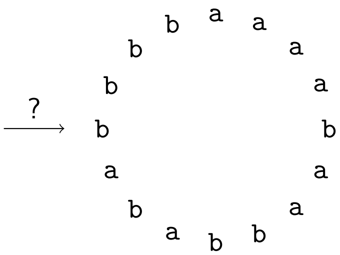
A de Bruijn word of order n is a circular word containing exactly one occurrence of all possible words of length n.

For instance, the words of length 4 over a binary alphabet are:

```
a a a a
a a a b
a a b a
a a b b
a b a a
a b a b
a b b a
a b b b
b a a a
b a a b
b a b a
b a b b
b b a a
b b a b
b b b a
b b b b
```

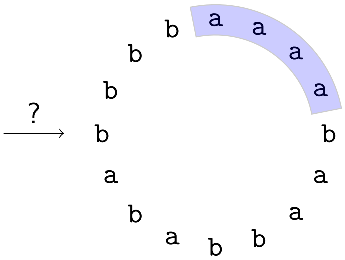
de Bruijn word of order *n*

a a a a
a a a b
a a b a
a a b b
a b a a
a b a b
a b b a
a b b b
b a a a
b a a b
b a b a
b a b b
b b a a
b b a b
b b b a
b b b b



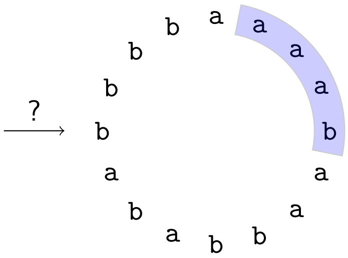
de Bruijn word of order n

a a a a
a a a b
a a b a
a a b b
a b a a
a b a b
a b b a
a b b b
b a a a
b a a b
b a b a
b a b b
b b a a
b b a b
b b b a
b b b b



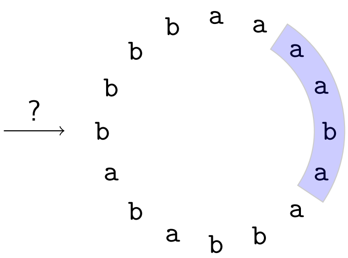
de Bruijn word of order n

a a a a
a a a b
a a b a
a a b b
a b a a
a b a b
a b b a
a b b b
b a a a
b a a b
b a b a
b a b b
b b a a
b b a b
b b b a
b b b b



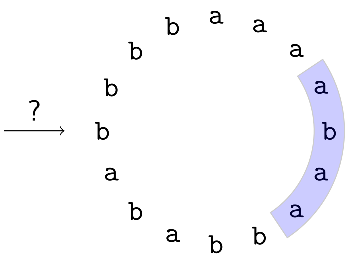
de Bruijn word of order n

a a a a
a a a b
a a b a
a a b b
a b a a
a b a b
a b b a
a b b b
b a a a
b a a b
b a b a
b a b b
b b a a
b b a b
b b b a
b b b b



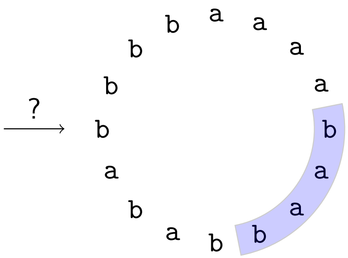
de Bruijn word of order *n*

a a a a
a a a b
a a b a
a a b b
a b a a
a b a b
a b b a
a b b b
b a a a
b a a b
b a b a
b a b b
b b a a
b b a b
b b b a
b b b b



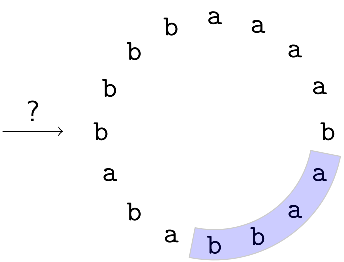
de Bruijn word of order n

a a a a
a a a b
a a b a
a a b b
a b a a
a b a b
a b b a
a b b b
b a a a
b a a b
b a b a
b a b b
b b a a
b b a b
b b b a
b b b b



de Bruijn word of order n

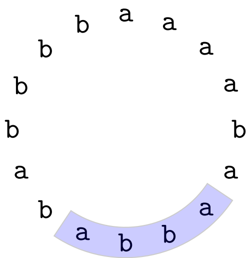
a a a a
a a a b
a a b a
a a b b
a b a a
a b a b
a b b a
a b b b
b a a a
b a a b
b a b a
b a b b
b b a a
b b a b
b b b a
b b b b



de Bruijn word of order n

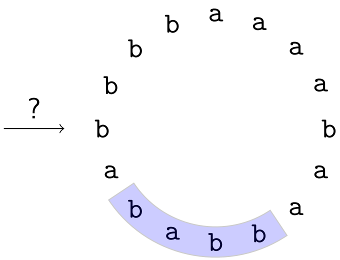
a a a a
a a a b
a a b a
a a b b
a b a a
a b a b
a b b a
a b b b
b a a a
b a a b
b a b a
b a b b
b b a a
b b a b
b b b a
b b b b

$\xrightarrow{?}$



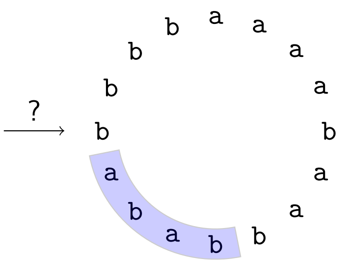
de Bruijn word of order n

a a a a
a a a b
a a b a
a a b b
a b a a
a b a b
a b b a
a b b b
b a a a
b a a b
b a b a
b a b b
b b a a
b b a b
b b b a
b b b b



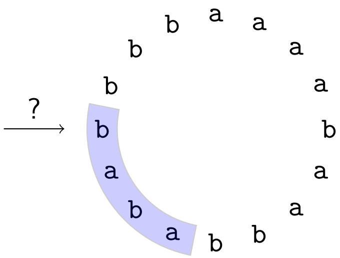
de Bruijn word of order n

a a a a
a a a b
a a b a
a a b b
a b a a
a b a b
a b b a
a b b b
b a a a
b a a b
b a b a
b a b b
b b a a
b b a b
b b b a
b b b b



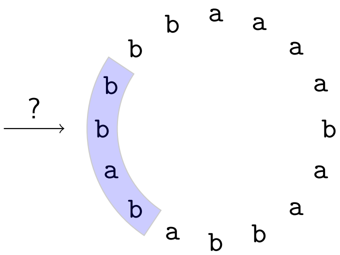
de Bruijn word of order n

a a a a
a a a b
a a b a
a a b b
a b a a
a b a b
a b b a
a b b b
b a a a
b a a b
b a b a
b a b b
b b a a
b b a b
b b b a
b b b b



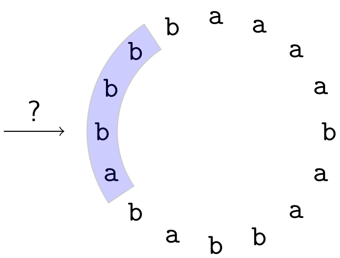
de Bruijn word of order n

a a a a
a a a b
a a b a
a a b b
a b a a
a b a b
a b b a
a b b b
b a a a
b a a b
b a b a
b a b b
b b a a
b b a b
b b b a
b b b b



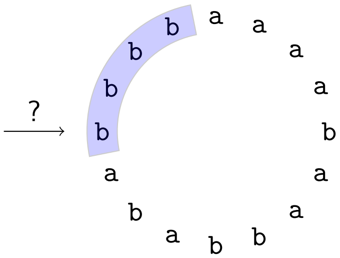
de Bruijn word of order n

a a a a
a a a b
a a b a
a a b b
a b a a
a b a b
a b b a
a b b b
b a a a
b a a b
b a b a
b a b b
b b a a
b b a b
b b b a
b b b b



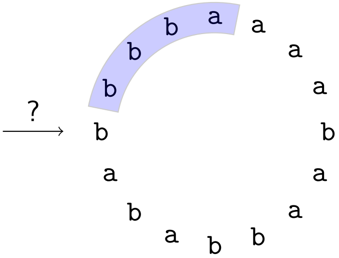
de Bruijn word of order *n*

a a a a
a a a b
a a b a
a a b b
a b a a
a b a b
a b b a
a b b b
b a a a
b a a b
b a b a
b a b b
b b a a
b b a b
b b b a
b b b b



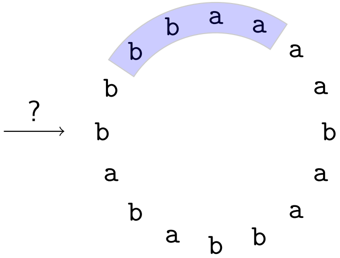
de Bruijn word of order n

a a a a
a a a b
a a b a
a a b b
a b a a
a b a b
a b b a
a b b b
b a a a
b a a b
b a b a
b a b b
b b a a
b b a b
b b b a
b b b b

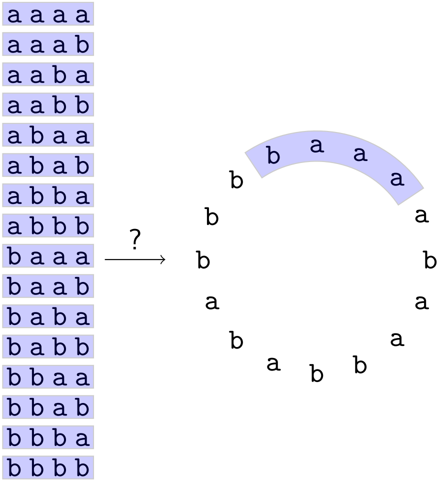


de Bruijn word of order n

a a a a
a a a b
a a b a
a a b b
a b a a
a b a b
a b b a
a b b b
b a a a
b a a b
b a b a
b a b b
b b a a
b b a b
b b b a
b b b b

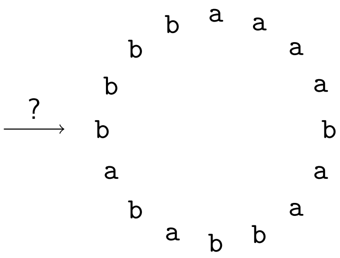


de Bruijn word of order n



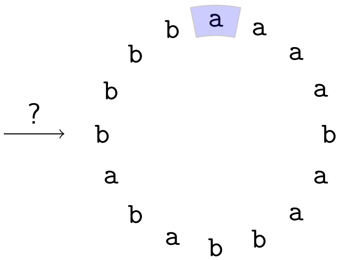
de Bruijn word of order n [Fredricksen et al,1978]

a
a a a b
a a b
a a b b
a b
a b b
a b b b
b



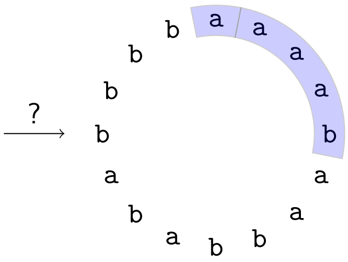
de Bruijn word of order n [Fredricksen et al,1978]

a
a a a b
a a b
a a b b
a b
a b b
a b b b
b

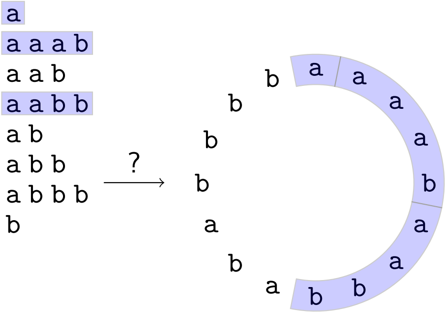


de Bruijn word of order n [Fredricksen et al,1978]

a
a a a b
a a b
a a b b
a b
a b b
a b b b
b

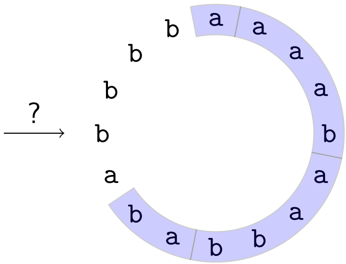


de Bruijn word of order n [Fredricksen et al,1978]



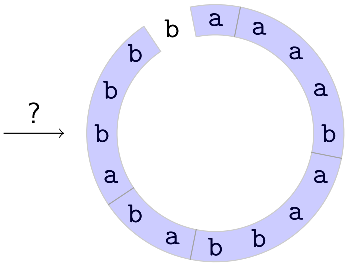
de Bruijn word of order n [Fredricksen et al,1978]

a
a a a b
a a b
a a b b
a b
a b b
a b b b
b



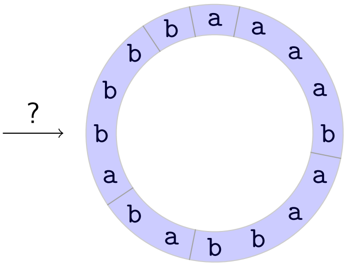
de Bruijn word of order *n* [Fredricksen et al,1978]

a
a a a b
a a b
a a b b
a b
a b b
a b b b
b

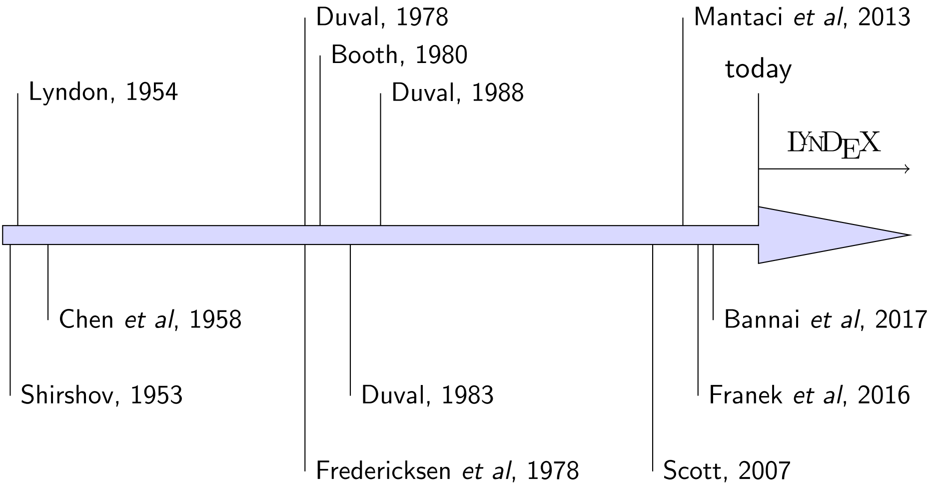


de Bruijn word of order n [Fredricksen et al,1978]

a
a a a b
a a b
a a b b
a b
a b b
a b b b
b



Back to the origins. . .



Conclusion

What I didn't talk about

- how Lyndon words can be useful for proving theorems
- Lyndon arrays
- Lyndon border arrays and Lyndon suffix arrays
- Lyndon words as convex envelops
- certainly many other things

Conclusion

Re "Lyndon words", I very much hope that they will some day be commonly known (also?) as "prime strings", because they are so fundamentally important.

D. E. Knuth, Oct. 2023