KIT
Karlsruhe Institute of Technology

Text Indexing

Lecture 00: Course Overview

Florian Kurpicz

Organizational Matters

Lectures

- Monday 10:00–11:30 (50.34, -119)
- lecture only

Organizational Matters

Lectures

- Monday 10:00–11:30 (50.34, -119)
- lecture only

Project (mandatory)

- topics will be handed out 08.11.2021
- coding project and small presentation
- 20 % of the final grade

Organizational Matters

Lectures

- Monday 10:00–11:30 (50.34, -119)
- lecture only

Project (mandatory)

- topics will be handed out 08.11.2021
- coding project and small presentation
- 20 % of the final grade

Oral Exam

- 20 minutes
- 80 % of the final grade
- pizza marks content not relevant for exam 🍕

Organizational Matters

Lectures

- Monday 10:00–11:30 (50.34, -119)
- lecture only

Project (mandatory)

- topics will be handed out 08.11.2021
- coding project and small presentation
- 20 % of the final grade

Oral Exam

- 20 minutes
- 80 % of the final grade
- pizza marks content not relevant for exam 🍕

Office Hours (Room 210)

- Monday 13:45–14:45 (lecture period)
- by appointment (otherwise)

Materials

Slides

- published after the lecture (`https://algo2.iti.kit.edu/4198.php`)

Videos

- will be published (with ≥ 1 week delay)

Materials

Slides

- published after the lecture (`https://algo2.iti.kit.edu/4198.php`)

Videos

- will be published (with $\geq$ 1 week delay)

Additional Material

- references to literature included
- books
  - Gonzalo Navarro. *Compact Data Structures - A Practical Approach*. Cambridge University Press, 2016. ISBN: 978-1-10-715238-0
  - Enno Ohlebusch. *Bioinformatics Algorithms: Sequence Analysis, Genome Rearrangements, and Phylogenetic Reconstruction*. Oldenbusch Verlag, 2013. ISBN: 978-3000413162
- most likely no script

Content

Fundamentals

- tries
- suffix tree
- suffix array
- longest common prefix array
- Burrows-Wheeler transform (BWT)
- wavelet tree (+ bit vector rank/select)
- FM-index

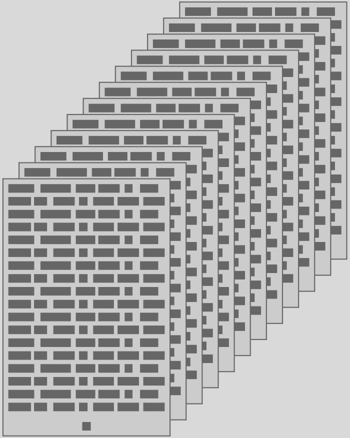
Compressed Indices

- compressing the BWT and wavelet trees
- Lempel-Ziv 77/78 compression
- LZ compression vs. BWT compression
- compressed suffix trees and suffix arrays
- r-index

Additional Topics

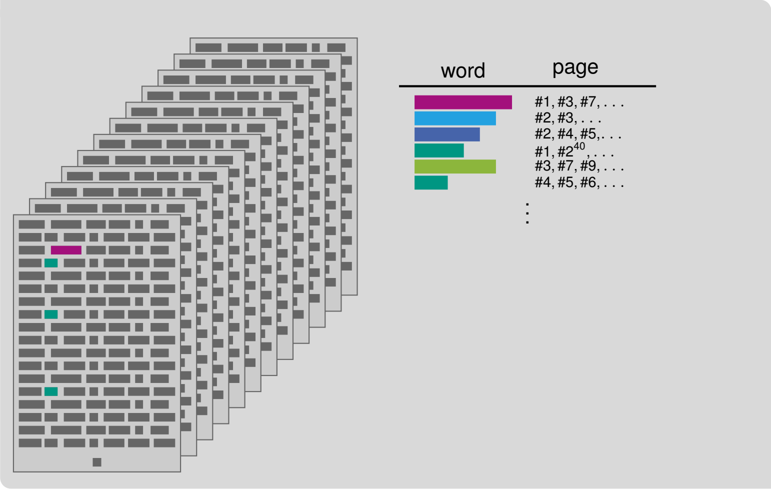
- parallel construction
- different query types

Motivation for Text Indices



- collection of text
- scanning not feasible

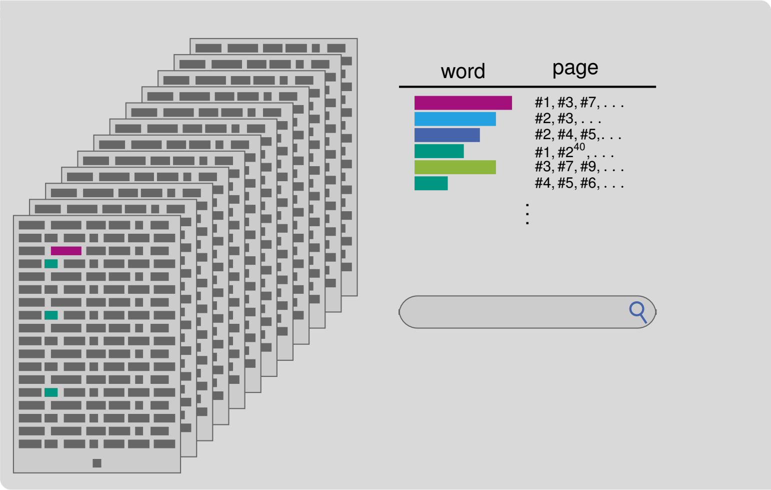
Motivation for Text Indices



word	page
🟪	#1, #3, #7, . . .
🟦	#2, #3, . . .
🟦	#2, #4, #5,. . .
🟩	#1, #2^{40}, . . .
🟩	#3, #7, #9, . . .
🟩	#4, #5, #6, . . .
	⋮

- collection of text
- scanning not feasible
- inverted index (word based)

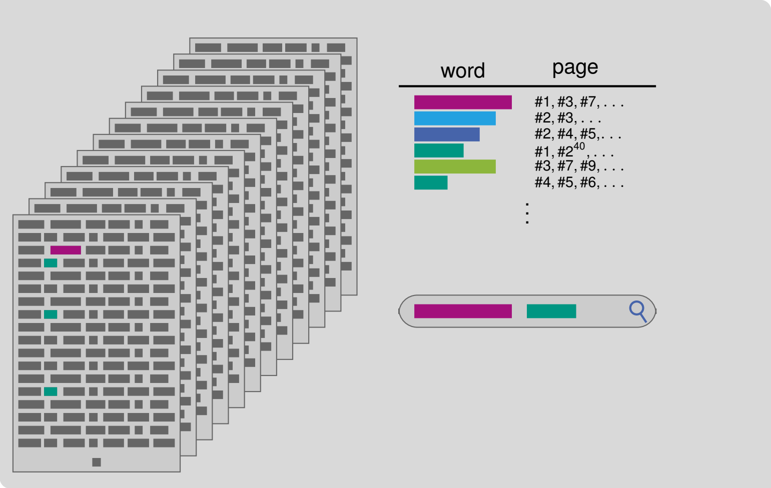
Motivation for Text Indices



word	page
🟪	#1, #3, #7, . . .
🟦	#2, #3, . . .
🟦	#2, #4, #5,. . .
🟩	#1, #2^{40}, . . .
🟩	#3, #7, #9, . . .
🟩	#4, #5, #6, . . .
⋮	

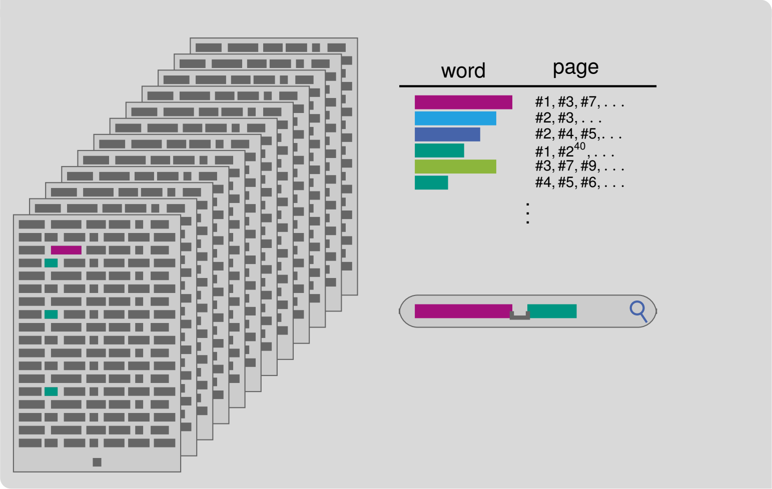
- collection of text
- scanning not feasible
- inverted index (word based)

Motivation for Text Indices



- collection of text
- scanning not feasible
- inverted index (word based)

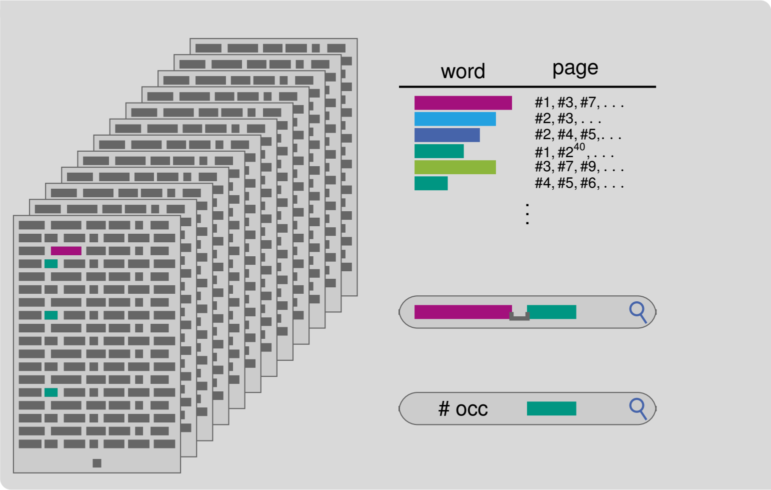
Motivation for Text Indices



word	page
🟪	#1, #3, #7, . . .
🟦	#2, #3, . . .
🟦	#2, #4, #5,. . .
🟩	#1, $\#2^{40}$, . . .
🟩	#3, #7, #9, . . .
🟩	#4, #5, #6, . . .
	⋮

- collection of text
- scanning not feasible
- inverted index (word based)
- phrase search

Motivation for Text Indices

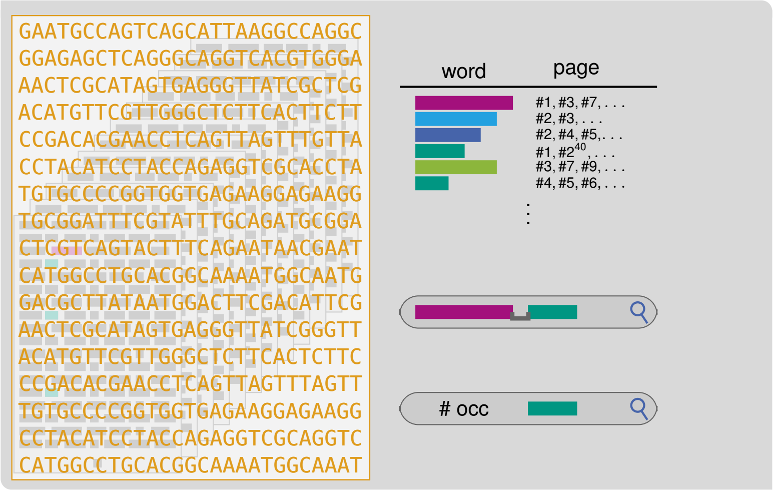


word page

#1, #3, #7, . . .
#2, #3, . . .
#2, #4, #5,. . .
#1, #2^{40}, . . .
#3, #7, #9, . . .
#4, #5, #6, . . .
⋮

- collection of text
- scanning not feasible
- inverted index (word based)
- phrase search
- counting queries

Motivation for Text Indices

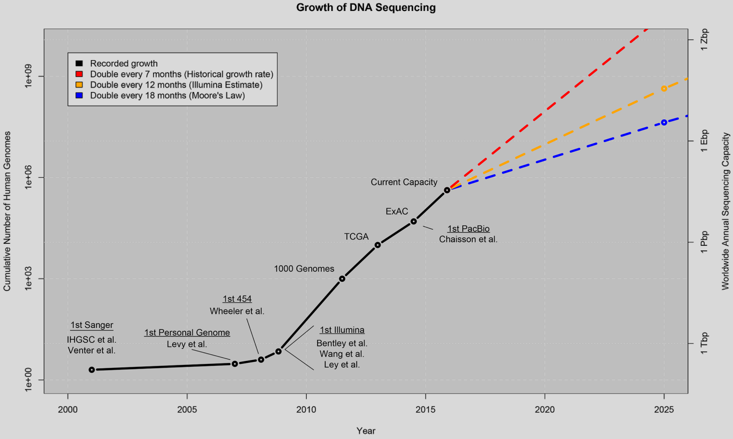

GAATGCCAGTCAGCATTAAGGCCAGGC
GGAGAGCTCAGGGCAGGTCACGTGGGA
AACTCGCATAGTGAGGGTTATCGCTCG
ACATGTTCGTTGGGCTCTTCACTTCTT
CCGACACGAACCTCAGTTAGTTTGTTA
CCTACATCCTACCAGAGGTCGCACCTA
TGTGCCCCGGTGGTGAGAAGGAGAAGG
TGCGGATTTCGTATTTGCAGATGCGGA
CTCGTCAGTACTTTCAGAATAACGAAT
CATGGCCTGCACGGCAAAATGGCAATG
GACGCTTATAATGGACTTCGACATTCG
AACTCGCATAGTGAGGGTTATCGGGTT
ACATGTTCGTTGGGCTCTTCACTCTTC
CCGACACGAACCTCAGTTAGTTTAGTT
TGTGCCCCGGTGGTGAGAAGGAGAAGG
CCTACATCCTACCAGAGGTCGCAGGTC
CATGGCCTGCACGGCAAAATGGCAAAT
word
page
#1, #3, #7, . . .
#2, #3, . . .
#2, #4, #5,. . .
#1, #2^40, . . .
#3, #7, #9, . . .
#4, #5, #6, . . .
⋮
# occ

- collection of text
- scanning not feasible
- inverted index (word based)
- phrase search
- counting queries
- what if there are no “words”

Why Texts?

Text is Everywhere

- Text-based Information
  - Wikipedia
  - dblp
  - books
  - news articles
  - code
- Very Important in Bioinformatics
  - DNA
  - proteins



[Ste+15]

Preliminaries (1/2)

Definition: Text

- let Σ be an **alphabet**
- $T \in \Sigma^\star$ is a text
- $|T| = n$ is the length of the string
- $T = T[1]T[2]\ldots T[n]$

Preliminaries (1/2)

Definition: Text

- let Σ be an **alphabet**
- $T \in \Sigma^\star$ is a text
- $|T| = n$ is the length of the string
- $T = T[1]T[2] \dots T[n]$

Definition: Alphabet Types

- **constant size alphabet**: finite set not depending on n
- **integer alphabet**: alphabet is $\{1, \dots, \sigma\}$ and fits into constant number of words
- **finite alphabets**: alphabet of finite size

Preliminaries (2/2)

Definition: Substring, Prefix, and Suffix

Given a text $T = T[1]T[2]\dots T[n]$ of length n:

- $T[i..j] = T[i]\dots T[j]$ is called a **substring**,

a	b	b	a	a	b	b	a	$

Preliminaries (2/2)

Definition: Substring, Prefix, and Suffix

Given a text $T = T[1]T[2]\dots T[n]$ of length n:

- $T[i..j] = T[i]\dots T[j]$ is called a **substring**,

a	b	b	a	a	b	b	a	$

- $T[1..i]$ is called a **prefix**, and

a	b	b	a	a	b	b	a	$

Preliminaries (2/2)

Definition: Substring, Prefix, and Suffix

Given a text $T = T[1]T[2]\dots T[n]$ of length n:

- $T[i..j] = T[i]\dots T[j]$ is called a **substring**,

a	b	**b**	**a**	**a**	**b**	b	a	$

- $T[1..i]$ is called a **prefix**, and

a	**b**	**b**	a	a	b	b	a	$

- $T[i..n]$ is called a **suffix** of T.

a	b	b	a	a	**b**	**b**	**a**	**$**

Preliminaries (2/2)

Definition: Substring, Prefix, and Suffix

Given a text $T = T[1]T[2]\dots T[n]$ of length n:

- $T[i..j] = T[i]\dots T[j]$ is called a **substring**,

a	b	b	a	a	b	b	a	$

- $T[1..i]$ is called a **prefix**, and

a	b	b	a	a	b	b	a	$

- $T[i..n]$ is called a **suffix** of T.

a	b	b	a	a	b	b	a	$

Sentinel for Simplicity

Given a text T of length n over an alphabet Σ.

- we assume that $T[n] = \$$ with
- $\$ \notin \Sigma$ and $\$ < \alpha$ for all $\alpha \in \Sigma$

Preliminaries (2/2)

Definition: Substring, Prefix, and Suffix

Given a text $T = T[1]T[2]\dots T[n]$ of length n:

- $T[i..j] = T[i]\dots T[j]$ is called a **substring**,

a	b	b	a	a	b	b	a	$

- $T[1..i]$ is called a **prefix**, and

a	b	b	a	a	b	b	a	$

- $T[i..n]$ is called a **suffix** of T.

a	b	b	a	a	b	b	a	$

Sentinel for Simplicity

Given a text T of length n over an alphabet Σ.

- we assume that $T[n] = \$$ with
- $\$ \notin \Sigma$ and $\$ < \alpha$ for all $\alpha \in \Sigma$

Preliminaries (2/2)

Definition: Substring, Prefix, and Suffix

Given a text $T = T[1]T[2]\dots T[n]$ of length n:

- $T[i..j] = T[i]\dots T[j]$ is called a **substring**,

a	b	b	a	a	b	b	a	$

- $T[1..i]$ is called a **prefix**, and

a	b	b	a	a	b	b	a	$

- $T[i..n]$ is called a **suffix** of T.

a	b	b	a	a	b	b	a	$

Sentinel for Simplicity

Given a text T of length n over an alphabet Σ.

- we assume that $T[n] = \$$ with
- $\$ \notin \Sigma$ and $\$ < \alpha$ for all $\alpha \in \Sigma$
- otherwise, suffix can be prefix of another suffix

1	2	3	4	5	6	7	8
a	b	b	a	a	b	b	a

- $T[1..n] = \texttt{abbaabba}$ and $T[5..n] = \texttt{abba}$

Preliminaries (2/2)

Definition: Substring, Prefix, and Suffix

Given a text $T = T[1]T[2]\dots T[n]$ of length n:

- $T[i..j] = T[i]\dots T[j]$ is called a **substring**,

a	b	**b**	**a**	**a**	**b**	b	a	$

- $T[1..i]$ is called a **prefix**, and

a	**b**	**b**	a	a	b	b	a	$

- $T[i..n]$ is called a **suffix** of T.

a	b	b	a	a	**b**	**b**	**a**	**$**

Sentinel for Simplicity

Given a text T of length n over an alphabet Σ.

- we assume that $T[n] = \$$ with
- $\$ \notin \Sigma$ and $\$ < \alpha$ for all $\alpha \in \Sigma$
- otherwise, suffix can be prefix of another suffix

1	2	3	4	5	6	7	8
a	b	b	a	a	b	b	a

- $T[1..n] = \texttt{abbaabba}$ and $T[5..n] = \texttt{abba}$

Definition: Prefix-Free

A string is **prefix-free** if no suffix is a prefix of another suffix

PINGO

Bibliography

[Nav16] Gonzalo Navarro. *Compact Data Structures - A Practical Approach*. Cambridge University Press, 2016. ISBN: 978-1-10-715238-0.

[Ohl13] Enno Ohlebusch. *Bioinformatics Algorithms: Sequence Analysis, Genome Rearrangements, and Phylogenetic Reconstruction*. Oldenbusch Verlag, 2013. ISBN: 978-3000413162.

[Ste+15] Zachary D Stephens., Skylar Y. Lee, Faraz Faghri, Roy H. Campbell, Chengxiang Zhai, Miles J. Efron, Ravishankar Iyer, Michael C. Schatz, Saurabh Sinha, and Gene E. Robinson. “Big Data: Astronomical or Genomical?” In: *PLOS Biology* 13.7 (July 2015), pages 1–11. DOI: `10.1371/journal.pbio.1002195`.