1.	Record Nr.	UNINA9910416139803321
	Autore	Mailund Thomas
	Titolo	String Algorithms in C : Efficient Text Representation and Search / / by Thomas Mailund
	Pubbl/distr/stampa	Berkeley, CA : , : Apress : , : Imprint : Apress, , 2020
	ISBN	9781484259207 1484259203
	Edizione	[1st ed. 2020.]
	Descrizione fisica	1 online resource (299 pages)
	Disciplina	005.1
	Soggetti	Programming languages (Electronic computers) Algorithms Programming Languages, Compilers, Interpreters Algorithm Analysis and Problem Complexity
	Lingua di pubblicazione	Inglese
	Formato	Materiale a stampa
	Livello bibliografico	Monografia
	Nota di bibliografia	Includes bibliographical references.
	Nota di contenuto	1. Introduction -- 2. Classical Algorithms for Exact Search -- 3. Suffix Trees -- 4. Suffix Arrays -- 5. Approximate Search -- 6. Conclusions -- Appendix A: Vectors -- Appendix B: Lists -- Appendix C: Queues.
	Sommario/riassunto	Implement practical data structures and algorithms for text search and discover how it is used inside other larger applications. This unique in-depth guide explains string algorithms using the C programming language. String Algorithms in C teaches you the following algorithms and how to use them: classical exact search algorithms; tries and compact tries; suffix trees and arrays; approximative pattern searches; and more. In this book, author Thomas Mailund provides a library with all the algorithms and applicable source code that you can use in your own programs. There are implementations of all the algorithms presented in this book so there are plenty of examples. You'll understand that string algorithms are used in various applications such as image processing, computer vision, text analytics processing from data science to web applications, information retrieval from databases, network security, and much more. You will: Use classical exact search algorithms including naive search, borders/border search, Knuth-Morris-Pratt, and Boyer-Moor with or without Horspool Search in trees,

use tries and compact tries, and work with the Aho-Carasick algorithm
Process suffix trees including the use and development of McCreight's
algorithm Work with suffix arrays including binary searches; sorting
naive constructions; suffix tree construction; skew algorithms; and the
Borrows-Wheeler transform (BWT) Deal with enhanced suffix arrays
including longest common prefix (LCP) Carry out approximative pattern
searches among suffix trees and approximative BWT searches .