1.	Record Nr.	UNINA9910819928003321
	Autore	Konheim Alan G. <1934->
	Titolo	Hashing in computer science : fifty years of slicing and dicing / / Alan G. Konheim
	Pubbl/distr/stampa	Hoboken, New Jersey : , : John Wiley & Sons, , c2010 [Piscataqay, New Jersey] : , : IEEE Xplore, , [2010]
	ISBN	1-118-03183-0 1-282-68627-5 9786612686276 0-470-63061-2 0-470-63060-4
	Edizione	[1st edition]
	Descrizione fisica	1 online resource (406 p.)
	Disciplina	005.8/2
	Soggetti	Hashing (Computer science) Cryptography Data encryption (Computer science) Computer security
	Lingua di pubblicazione	Inglese
	Formato	Materiale a stampa
	Livello bibliografico	Monografia
	Note generali	Description based upon print version of record.
	Nota di bibliografia	Includes bibliographical references and index.
	Nota di contenuto	PREFACE -- PART I: MATHEMATICAL PRELIMINARIES -- 1. Counting -- 1.1: The Sum and Product Rules -- 1.2: Mathematical Induction -- 1.3: Factorial -- 1.4: Binomial Coefficients -- 1.5: Multinomial Coefficients -- 1.6: Permutations -- 1.7: Combinations -- 1.8: The Principle of Inclusion-Exclusion -- 1.9: Partitions -- 1.10: Relations -- 1.11: Inverse Relations -- Appendix 1: Summations Involving Binomial Coefficients -- 2. Recurrence and Generating Functions -- 2.1: Recursions -- 2.2: Generating Functions -- 2.3: Linear Constant Coefficient Recursions -- 2.4: Solving Homogeneous LCCRs Using Generating Functions -- 2.5: The Catalan Recursion -- 2.6: The Umbral Calculus -- 2.7: Exponential Generating Functions -- 2.8: Partitions of a Set: The Bell and Stirling Numbers -- 2.9: Rouche's Theorem and the Lagrange's Inversion Formula -- 3. Asymptotic Analysis -- 3.1: Growth Notation for Sequences -- 3.2: Asymptotic Sequences and Expansions -- 3.3: Saddle Points -- 3.4: Laplace's Method -- 3.5: The Saddle Point

Method -- 3.6: When Will the Saddle Point Method Work? -- 3.7: The Saddle Point Bounds -- 3.8: Examples of Saddle Point Analysis -- 4. Discrete Probability Theory -- 4.1: The Origins of Probability Theory -- 4.2: Chance Experiments, Sample Points, Spaces, and Events -- 4.3: Random Variables -- 4.4: Moments - Expectation and Variance -- 4.5: The Birthday Paradox -- 4.6: Conditional Probability and Independence -- 4.7: The Law of Large Numbers (LLN) -- 4.8: The Central Limit Theorem (CLT) -- 4.9: Random Processes and Markov Chains -- 5. Number Theory and Modern Algebra -- 5.1: Prime Numbers -- 5.2: Modular Arithmetic and the Euclidean Algorithm -- 5.3: Modular Multiplication -- 5.4: The Theorems of Fermat and Euler -- 5.5: Fields and Extension Fields -- 5.6: Factorization of Integers -- 5.7: Testing Primality -- 6. Basic Concepts of Cryptography -- 6.1: The Lexicon of Cryptography -- 6.2: Stream Ciphers -- 6.3: Block Ciphers -- 6.4: Secrecy Systems and Cryptanalysis -- 6.5: Symmetric and Two-Key Cryptographic Systems.

6.6: The Appearance of Public Key Cryptographic systems -- 6.7: A Multitude of Keys -- 6.8: The RSA Cryptosystem -- 6.9: Does PKC Solve the Problem of Key Distribution? -- 6.10: Elliptic Groups Over the Reals -- 6.11: Elliptic Groups Over the Field Zm,2 -- 6.12: Elliptic Group Cryptosystems -- 6.13: The Menezes-Vanstone Elliptic Curve Cryptosystem -- 6.14: Super-Singular Elliptic Curves -- PART II: HASHING FOR STORAGE: DATA MANAGEMENT -- 7. Basic Concepts -- 7.1: Overview of the Records Management Problem -- 7.2: A Simple Storage Management Protocol: Plain Vanilla Chaining -- 7.3: Record-Management with Sorted Keys -- 8. Hash Functions -- 8.1: The Origin of Hashing -- 8.2: Hash Tables -- 8.3: A Statistical Model for Hashing -- 8.4: The Likelihood of Collisions -- 9. Hashing Functions: Examples and Evaluation -- 9.1: Overview: The Tradeoff of Randomization Versus Computational Simplicity -- 9.2: Some Examples of Hashing Functions -- 9.3: Performance of Hash Functions: Formulation -- 9.4: The X2-Test -- 9.5: Testing a Hash Function -- 9.6: The McKenzie et al. Results -- 10. Record Chaining with Hash Tables -- 10.1: Separate Chaining of Records -- 10.2: Analysis of Separate Chaining Hashing Sequences and the Chains They Create -- 10.3: A Combinatorial Analysis of Separate Chaining -- 10.4: Coalesced Chaining -- 10.5: The Pittel-Yu Analysis of EICH Coalesced Chaining -- 10.6: To Separate or to Coalesce; and Which Version? That Is the Question -- 11. Perfect Hashing -- 11.1: Overview -- 11.2: Chichelli's Construction -- 12. The Uniform Hashing Model -- 12.1: An Idealized Hashing Model -- 12.2: The Asymptotics of Uniform Hashing -- 12.3: Collision-Free Hashing -- 13. Hashing with Linear Probing -- 13.1: Formulation and Preliminaries -- 13.2: Performance Measures for LP Hashing -- 13.3: All Cells Other than HTn-1 in the Hash-Table of n Cells are Occupied -- 13.4: m-Keys Hashed into a Hash Table of n Cells Leaving Cell HTn-1 Unoccupied -- 13.5: The Probability Distribution for the Length of a Search.

13.6: Asymptotics -- 13.7: Hashing with Linear Open Addressing: Coda -- 13.8: A Possible Improvement to Linear Probing -- 14. Double Hashing -- 14.1: Formulation of Double Hashing -- 14.2: Progressions and Strides -- 14.3: The Number of Progressions Which Fill a Hash-Table Cell -- 14.3.1: Progression Graphs -- 14.4: Dominance -- 14.5: Insertion-Cost Bounds Relating Uniform and Double Hashing -- 14.6: UsuallyDoubleHash -- 14.7: The UDH Chance Experiment and the Cost to Insert the Next Key by Double Hashing -- 14.8: Proof of Equation (14.12a) -- 14.9: UsuallyDoubleHash -- 14.10: Proof of Equation (14.12b) -- 15. Optimum Hashing -- 15.1: The Ullman / Yao Framework -- 15.1.1: The Ullman / Yao Hashing Functions -- 15.1.2:

Ullman / Yao INSERT(k) and SEARCH(k) -- 15.1.3: The Ullman / Yao Statistical Model -- 15.2: The Rates at Which a Cell is Probed and Occupied -- 15.3: Partitions of (i)Scenarios, (i)Subscenarios, and Their Skeletons -- 15.3.1: (i)Subscenarios -- 15.3.2: Skeletons -- 15.4: Randomly Generated m-Scenarios -- 15.5: Bounds on Random Sums -- 15.6: Completing the Proof of Theorem 15.1 -- PART III: SOME NOVEL APPLICATIONS OF HASHING -- 16. Karp-Rabin String Searching -- 16.1: Overview -- 16.2: The Basic Karp-Rabin Hash-Fingerprint Algorithm -- 16.3: The Plain Vanilla Karp-Rabin Fingerprint Algorithm -- 16.4: Some Estimates on Prime Numbers -- 16.5: The Cost of False Matches in the Plain Vanilla Karp-Rabin Fingerprint Algorithm -- 16.6: Variations on the Plain Vanilla Karp-Rabin Fingerprint Algorithm -- 16.7: A Nonhashing Karp-Rabin Fingerprint -- 17. Hashing Rock and Roll -- 17.1: Overview of Audio Fingerprinting -- 17.2: The Basics of Fingerprinting Music -- 17.3: Haar Wavelet Coding -- 17.4: Min-Hash -- 17.5: Some Commercial Fingerprinting Products -- 18. Hashing in E-Commerce -- 18.1: The Varied Applications of Cryptography -- 18.2: Authentication -- 18.3: The Need for Certificates -- 18.4: Cryptographic Hash Functions -- 18.5: X.509 Certificates and CCIT Standardization.

18.6: The Secure Socket Layer (SSL) -- 18.7: Trust on the Web ... Trust No One Over 40! -- 18.8: MD5 -- 18.9: Criticism of MD5 -- 18.10: The Wang-Yu Collision Attack -- 18.11: Steven's Improvement to the Wang-Yu Collision Attack -- 18.12: The Chosen-Prefix Attack on MD5 -- 18.13: The Rogue CA Attack Scenario -- 18.14: The Secure Hash Algorithms -- 18.15: Criticism of SHA-1 -- 18.16: SHA-2 -- 18.17: What Now? -- Appendix 18: Sketch of the Steven's Chosen Prefix Attack -- 19. Hashing and the Secure Distribution of Digital Media -- 19.1: Overview -- 19.2: Intellectual Property (Copyrights and Patents) -- 19.3: Steganography -- 19.4: Boil, Boil, Toil ... and But First, Carefully Mix -- 19.5: Software Distribution Systems -- 19.6: Watermarks -- 19.7: An Image-Processing Technique for Watermarking -- 19.8: Using Geometric Hashing to Watermark Images -- 19.9: Biometrics and Hashing -- 19.10: The Dongle -- Appendix 19: Reed-Solomon and Hadamard Coding -- Exercises and Solutions -- INDEX.

Sommario/riassunto

Gain the Skills and Knowledge Needed to Understanding Data Security Systems A file of computer data is composed of records to each of which a key or identifier is associated. The key is used to search for the address of a desired record. When the file is a telephone directory, searching is easy - the key is the subscriber's name and the records are naturally arranged in alphabetic order. For data whose records are not easily alphabetized, a hash function is used to arithmetically derive from the key record's address. Hashing was invented during the design of the IBM 701 machine in the 1950s by Hans Peter Luhn. In the ensuing half century, the hashing concept has found a variety of applications. When combined with cryptography, hashing can be used to authenticate users in e-commerce on the Web. Professor Konheim is an authority on computer security and an early contributor to hashing technology. Based on courses taught by the author, this book unravels the complicated mathematics involved in hashing as it explains in detail the various hashing methods. It describes: . Techniques for audio fingerprinting, the automated recognition of music. The use of hashing in e-commerce to protect against identity theft. How hashing is used to inhibit the unlawful copying and distribution of music, video, software, books, and data Key points are reinforced in the sample problems and solutions provided; also included are an accompanying instructor's manual and extensive bibliography. Hashing in Computer Science is valuable reading for graduate students and researchers in mathematics,

cryptography, and security. It can be used as a textbook in senior and graduate courses on cryptography and others that employ cryptanalysis, computer security, analysis of randomized and combinatorial algorithms, computer networks, compiler design, computational geometry, and theory of computation.