Index

2–3 and 2–3–4 trees, 5–59
9/11 Memorial, 11–28
123456791, 7–66
987654263, 7–66
∀ (universal quantifier), 3–42 ff.
absolute value, 2–7, 4–34, 4–36
abstract algebra, 7–46
abstraction, 4–2
ACM Code of Ethics and Professional Conduct, 12–3
ACM Conference on Fairness, Accountability, and Transparency, 8–19
adjacency, see graphs
affirming the consequent, see fallacy
algorithmic bias, 2–62, 4–86, 4–87, 8–19, 12–3
algorithmic sentencing, 8–19
algorithms, 2–83 ff., see also randomized algorithms
asymptotic analysis, 6–22 ff.
brute force, 3–32, 5–17, 9–2, 9–73
divide and conquer, 6–60 ff., 6–68
dynamic programming, 5–17, 9–2, 9–73
greedy algorithms, 4–28, 9–20
recurrence relations, 6–42 ff.
time, space, and power, 6–32
Alice and Bob, 7–58 ff.
ambiguity
in natural language, 3–10, 3–11, 3–17
order of operations, 5–56, 8–7
order of quantification, 3–65 ff., 3–78
prefix-free/Huffman codes, 9–20
analysis (mathematics), 8–48
ancestors (in a tree), 11–63
and (∧), 3–7
anonymization, 10–20
antisymmetry, 8–26 ff.
approximate equality, 2–6
Ariane 5 rocket, 4–82
arithmetic mean, 4–50, 4–72
arithmetic series, 2–15, 5–14, 5–25
Arrow’s Theorem, 8–38
artificial intelligence, 12–3
computer vision, 11–38
game trees, 3–54, 9–52
assertions, 3–77, 5–20
associativity, 3–27, 5–59, 7–46
assuming the antecedent, see proofs
asymmetry, 8–26 ff.
asymptotics
analysis of algorithms, 6–22 ff.
asymptotic analysis, 6–4 ff.
asymptotic relationships viewed as relations, 8–31 ff.
best- and average-case running time, 6–29 ff.
divide and conquer, 6–61 ff.
O (Big O), 6–5 ff.
Ω, ω, and Θ, 6–10 ff.
polynomials, logs, and exponentials, 6–8 ff.
recurrence relations, 6–42 ff.
worst-case analysis, 6–23 ff.
avtomata, 8–59, 9–54
automated theorem proving, 4–31
average distance in a graph, 11–54
average-case analysis, see running time
AVL trees, 6–53 ff.
axiom of extensionality, 2–35
Bacon, Kevin, 4–46, 11–20
balanced binary search trees, 6–53
Bayes’ Rule, 10–44 ff.
begging the question, see fallacy
Bernoulli distribution, 10–16 ff., 10–60, 10–73
Bernoulli’s inequality, 5–26
betweenness, 8–15
BFS, see breadth-first search
bias, see algorithmic bias
biased coins, 10–17 ff.
bigrams, 10–48
bijection, 2–79, 9–35, 9–46
binary numbers, see integers
binary relation, see relations
Binary Search, see searching
binary search trees, see trees
binary symmetric channel, 10–44, 10–45
binary trees, see trees
binomial coefficients, see combinations
binomial distribution, 10–17 ff., 10–65
Binomial Theorem, 9–67 ff.
bipartite graphs, 11–21 ff.
complete bipartite graphs, 11–22
birthday paradox, 5–34, 10–68
Bitcoin, 12–2
bitmaps, 2–56
bits/bitstrings, 2–4, 2–51, 3–20, 4–5 ff., 7–58, 9–6, 9–37, 9–58 ff., 9–79
Bletchley Park, 9–75
blockchain, 12–2
Bloom filters, 10–54
Bob smells, 5–22
Booleans, 2–4, 3–6, see also logic

This material will be published by Cambridge University Press as Connecting Discrete Mathematics and Computer Science by David Liben-Nowell, and an older edition of the material was published by John Wiley & Sons, Inc as Discrete Mathematics for Computer Science. This pre-publication version is free to view and download for personal use only. Not for re-distribution, re-sale, or use in derivative works. © David Liben-Nowell 2020–2021. This version was posted on September 8, 2021.
Index

bound (vs. free) variables, 3-46, 3-56
breadth-first search, 11-42 ff.
finding cycles, 11-59
brute force, see algorithms
Bubble Sort, see sorting
Buffon’s needle, 10-76
bugs, 2-20, 4-82, 5-20, 11-34
butterfly ballots, 8-38

C (programming language), 3-34, 3-56, 5-47
Caesar Cipher, see cryptography
cardinality, 2-28–2-29, 9-4 ff.
infinite, 9-46
Carmichael numbers, 7-51, 7-53, 7-56
Cartesian plane, 2-50
Cartesian product (×), 2-48
catchphrase, 8-40, 11-81
Cauchy sequences, 8-48
ceiling, 2-7
cellular automata, 9-54
Chain Rule (probability), 10-43 ff.
change of index, 2-15
checkers, 3-54, 4-48, 9-31
checksum, 4-4, 4-15
chess, 2-48, 3-54, 5-25, 9-15, 9-30, 11-40
children (in a tree), 11-62
Chinese Remainder Theorem, 7-30 ff.
circle packing, 4-21
circuits, 3-2, 3-20
printing and planar graphs, 11-24
representing logical propositions, 3-27, 3-37
using nand gates, 4-57
class-size paradox, 10-61
cliques, 11-19 ff.
closure, 7-46, 8-33 ff.
clustering, 2-42
coursening equivalence relations, 8-48 ff.
codomain (of a function), 2-72
collaboration networks, 11-20
collaborative filtering, 2-45
combinations, 9-57 ff.
k-combinations, 9-60 ff.

Binomial Theorem, 9-67 ff.
Pascal’s identity, 9-66, 9-71
Pascal’s triangle, 9-70
combinatorial proof, 9-64 ff.
commutativity, 2-58, 3-27, 3-66, 5-59, 7-46
comparability, see partial orders
comparison-based sorting, see sorting
complement (of a set), 2-33
complete graphs, 11-19 ff.
complexity, see computational complexity
composite numbers, 2-13, see also prime numbers
composition
of functions, 2-76, 8-13
of relations, 8-9, 8-30
compression
entropy and compressibility, 10-24
Huffman coding, 9-20, 9-48
impossibility of lossless compression, 9-48
lossy vs. lossless, 9-48
quantization of images, 2-71, 2-87
URL shortening, 9-9
computability, 4-62
computational biology
gene rearrangements, 3-76, 9-54
motifs in gene networks, 11-18
computational complexity
and cryptography, 7-65
complexity classes, 6-35
graph isomorphism, 11-17
input size, 7-8
P vs. NP, 3-32
regular languages, 8-40, 8-59
computational geometry, 2-66
computational linguistics, see natural language processing
computer architecture, 3-28 ff., 4-58
and running times, 6-23
Moore’s Law, 6-16
power consumption, 6-32
representation of numbers, 2-20
computer graphics
hidden-surface removal, 8-61
morphing, 2-68
rotation matrices, 2-63
triangulation, 5-36
computer security, 7-65–7-67
computer vision, 11-38
computing networking, 9-22
conditional expectation, 10-71 ff.
conditional independence, 10-42
conditional probability, 10-36 ff.
Bayes’ Rule, 10-44
Chain Rule, 10-43
Law of Total Probability, 10-43
Condorcet paradox, 8-38
congruences (modular), 7-8 ff., 7-31 ff., 8-47
conjunctive normal form, 3-29 ff., 4-53 ff., 5-52 ff.
connectivity (in graphs), 11-36 ff.
connected component, 11-36 ff.
reachability, 11-38 ff.
constructive proofs, 4-41
constructivism, 4-42
decomposition, 10-43
context-free grammar, 5-56
contradiction, 3-23
contrapositive, 3-25, 4-36, see also proofs
converse, 3-25
Cook–Levin Theorem, 3-32
correlation, 10-30
correlation vs. causation, 4-81
positive and negative, 10-33
countable sets, 9-46
counterexamples, 4-40 ff.
counting
Binomial Theorem, 9-67 ff.
combinations, 9-57 ff.
combinatorial proofs, 9-64 ff.
combining products and sums, 9-17 ff.
Division Rule, 9-38 ff.
double counting, 9-10 ff.
Generalized Product Rule, 9-14 ff.
inclusion–exclusion, 9-10 ff.
for 3+ sets, 9-13
Mapping Rule, 9-34 ff.
degree (in a graph), 11-8 ff., 11-10
degree distribution, 11-8 ff.
degree, 5-3, 11-8 ff.
regular graphs, 11-22
degree (of a polynomial), 2-82
density (of a graph), 6-20, 11-32
denying the hypothesis, see fallacy
dependent events, 10-30 ff.
depth-first search, 11-46 ff.
Descartes, René, 2-48
descendants (in a tree), 11-63
deterministic finite automata, 8-59
depth-first search
DFS, see depth-first search
diagonalization, 9-46
diameter (of a graph), 11-53
differential privacy, 10-20
Diffie–Hellman key exchange, 7-67
disconnected, see connectivity in graphs
disjoint sets, 2-37, 4-20
disjunctive normal form, 3-29 ff., 4-53 ff., 5-52 ff.
distance, see also metrics
Euclidean, see Euclidean distance
Hamming, see Hamming distance
in a graph, 11-41 ff.
Manhattan, see Manhattan distance
minimum distance of a code, 4-8 ff.
divide and conquer, see algorithms
divisibility, 2-12, 2-74, 5-18, 8-53
and modular arithmetic, 7-9 ff.
common divisors, 7-11 ff.
divisibility rules, 3-20, 4-33, 4-50, 7-19
Division Theorem, 7-4
division, see mod
in \( \mathbb{Z}_n \), 7-44
Division Rule, 9-38 ff.
domain (of a function), 2-72
dot product, 2-53 ff.
Dunbar’s number, 11-30
dynamic programming, see algorithms
dynamic scope, 3-56
\( \exists \) (existential quantifier), 3-42 ff.
\( (base of natural logarithm), 2-11
edges, see graphs
efficiency, see running time, see also
computational complexity
ELIZA, 12-3
Ermacs, 9-29, 12-3
empty set, 2-32
Enigma Machine, 9-75
entropy, 10-24
equivalence relations, 8-45 ff.
equivalence classes, 8-47
refinements and coarsenings, 8-48
Eratosthenes, 7-22, 7-40
Erdős numbers, 4-46
Erdős, Paul, 4-46, 11-20
error-correcting codes, 4-6 ff.
Golay code, 4-28
Hamming code, 4-15 ff., 9-33
messages and codewords, 4-6 ff.
minimum distance and rate, 4-8 ff.
Reed–Solomon codes, 4-23, 7-38
repetition code, 4-13 ff.
upper bounds on rates, 4-19
error-detecting codes, 4-6 ff.
credit card numbers, 4-4, 4-25
UPC, 9-51
eths, 4-86, 5-22, 6-66, 8-19, 10-20,
12-2–12-4
Euclid, 4-60, 7-12
Euclidean algorithm, 7-12, 7-26
efficiency, 7-15, 7-19
Extended Euclidean algorithm, 7-27
Euclidean distance, 2-66, 4-72
Euler’s Theorem, 7-56
even numbers, 2-13, 4-38
evenly divides, see divisibility
events (probability), 10-8 ff.
correlated, 10-30
independent events, 10-30 ff.
exclusive or (\( \oplus \)), 2-13, 3-10 ff., 4-15 ff.
existential quantifier (\( \exists \)), 3-42 ff.
extpectation, 10-60 ff.
average-case analysis of algorithms, 6-29 ff.
Index

conditional expectation, 10-71 ff.
coupon collector problem, 10-84
development from expectation, 10-72 ff.
Markov’s inequality, 10-84
Law of Total Expectation, 10-72
linearity of expectation, 10-64 ff.
exponentials, 2-8 ff., 5-59
asymptotics, 6-9 ff.
multiplicative modular, 7-19
EXPSPACE (complexity class), 6-35
EXPTIME (complexity class), 6-35
Extended Euclidean algorithm, 7-27

facial recognition, 4-86, 4-87, 8-19, 12-3
factorial, 4-30–4-31, 5-18, 6-43, 6-45, 9-16, 9-27
Stirling’s approximation, 9-80
factors, see divisibility, see also prime factorization
fallacy, 4-75 ff.
affirming the consequent, 4-78
begging the question, 4-79
denying the hypothesis, 4-78
false dichotomy, 4-35, 4-79
proving true, 4-77
false dichotomy, see fallacy
Fast Fourier transform, 2-9
fencepost error, 11-34
Fermat pseudoprime, 7-51
Fermat’s Last Theorem, 7-48
Fermat’s Little Theorem, 7-48 ff.
Fermat–Euler Theorem, 7-56
Fibonacci numbers, 2-68, 5-41, 6-44, 6-49–6-52, 6-55, 9-80
algorithms, 6-58
and the Euclidean algorithm, 7-19
filter, 2-40
finite-state machines, 8-59
float (floating point number), 2-20, 6-23
floor, 2-7
Division Theorem, 7-4
for loops
analogy for \( \prod \), 2-19
analogy for \( \sum \), 2-14, 2-16
analogy for quantifiers, 3-45, 3-67, 3-78
forests, 11-60
spanning forests, 11-68
formal language theory, 8-40, see computational complexity
formal methods, 4-31, 8-32
Four Color Theorem, 4-48, 11-24
fractals, 5-2, 5-10–5-11, 5-25–5-26, 5-42
free (vs. bound) variables, 3-46, 3-56
frequency analysis, 10-34
functions, 2-70 ff.
algorithms, 2-83 ff.
characteristic function of a set, 8-7
composition, 2-76
domain/codomain, 2-72
growth rates, 6-4 ff.
function, 2-80
one-to-one/onto functions, 2-77 ff.
function/dependence, 2-73 ff.
viewed as relations, 8-12 ff.
visual representation, 2-75
vs. macros, 3-56
Fundamental Theorem of Arithmetic, 7-24
fuzzy logic, 3-18
Game of Life, 9-54
game trees, 3-54, 9-52
garbage collection, 6-33, 11-51
Gates, Bill, 3-76, 4-46
GCD, see greatest common divisor
GCHQ, 7-60
Generalized Product Rule, 9-14 ff.
geometric distribution, 10-18 ff., 10-64
growth rates, 6-4 ff.
geometric mean, 4-50, 4-72
geometric series, 2-15, 5-12 ff.
for recurrence relations, 6-61 ff.
infinite, 5-14
Gödel’s Incompleteness Theorem, 3-58
Goldbach’s conjecture, 3-4, 3-64, 3-78
golden ratio, 6-51, 6-55
grammars, 5-47, 5-56
graph drawing, 11-23, 11-28
graphs, 11-4 ff.
acyclic graphs, 11-59 ff.
adversion lists, 11-12 ff.
adversary matrices, 11-13 ff.
bipartite graphs, 11-21 ff.
breadth-first search, 11-42 ff.
complete graphs, 11-19 ff.
connected components, 11-36 ff.
connectivity, 11-36 ff.
cycles, 11-57 ff.
data structures, 11-11, 11-12 ff.
degree, 11-8, 11-10 ff.
Handshaking Lemma, 11-9
regular graphs, 11-22
density, 11-32
depth-first search, 11-46 ff.
forests, 11-60
isomorphism, 11-16 ff.
matching, 9-42, 9-53, 9-75, 11-23, 11-71
neighborhoods, 11-7 ff., 11-10 ff.
paths, 11-34 ff.
shortest paths, 11-41 ff.
planar graphs, 11-23 ff.
shortest paths
Dijkstra’s algorithm, 11-80 ff.
simple graphs, 11-6
subgraphs, 11-17 ff.
trees, see trees
undirected vs. directed, 11-4 ff.
weighted graphs, 11-79 ff.
Dijkstra’s algorithm, 11-80 ff.
grayscale, 2-2, 2-71, 3-38
greatest common divisor, 7-11 ff., see also Euclidean algorithm
H_n, see harmonic number
Halting Problem, 3-58, 4-64 ff., 4-70
Hamiltonian path, 11-54
Hamming code, 2-69, 4-15 ff., 9-52
number of valid codewords, 9-33
Hamming distance, 4-5
Handshaking Lemma, 11-9
harmonic number, 5-14–5-15, 5-26

This material will be published by Cambridge University Press as *Connecting Discrete Mathematics and Computer Science* by David Liben-Nowell, and an older edition of the material was published by John Wiley & Sons, Inc as *Discrete Mathematics for Computer Science*. This pre-publication version is free to view and download for personal use only. Not for re-distribution, re-sale, or use in derivative works. © David Liben-Nowell 2020–2021. This version was posted on September 8, 2021.
hashing, 2-85, 9-55, 10-3–10-5, 10-66, 10-84
Bloom filters, 10-54
chaining, 2-85
collisions, 2-85, 10-3 ff., 10-13, 10-28, 10-54, 10-67
and pairwise independence, 10-36
chaining, 10-3
clustering, 10-13, 10-28
double hashing, 10-29
linear probing, 10-13, 10-28
quadratic probing, 10-28
simple uniform hashing, 10-4
Hasse diagrams, 8-52
heaps, 2-88, 5-38, 5-59
heavy-tailed distribution, 11-26
height (of a tree), 11-63
Heron’s method, 2-22, 4-50
hidden-surface removal, 8-61
higher-order functions, 2-40, 3-73
Hopper, Grace Murray, 2-71, 4-82
Huffman coding, 9-20, 9-48
hypercube, 11-32
I (identity matrix), 2-56
idempotence, 3-27
identity
identity function, 2-80
identity matrix, 2-56
multiplicative identity, 7-44
of a binary operator, 3-19, 5-59
if and only if (⇔), 3-10 ff.
image (of a function), 2-73
image processing
blur filter, 2-22
dithering, 3-38
grayscale conversion, 2-2
quantization, 2-71
seam carving, 9-73
segmentation, 11-38
imaginary numbers, 2-9
implication (⇒), 3-8 ff.
in-degree, see degree
in-neighbor, see neighbors (in graphs)
inclusion–exclusion, 9-10 ff., 9-80
inclusive naming, 6-66
incomparability, 6-13, 8-50
incompleteness (logic), 3-58
inconsistency (logic), 3-58
independent events, 10-30 ff.
pairwise independence, 10-36
indicator random variables, 10-59
induction, see proofs
checklist for inductive proofs, 5-6
generating conjectures, 5-10
proofs about algorithms, 5-16 ff., 6-46 ff.
strengthening the inductive hypothesis, 5-53
infinite sequences, 2-15, 5-14
infix notation, 8-6
information retrieval, 2-61
information theory, 10-24, 10-44
injective functions, see one-to-one functions
Insertion Sort, see sorting
integers, 2-4 ff.
algorithms for arithmetic, 7-6, 7-18
efficiency, 7-8
division, see modular arithmetic
primes and composites, see prime numbers
recursive definition, 5-55
representation
binary numbers, 3-20, 5-7, 5-27, 5-40, 7-8, 7-16
different bases, 5-40, 7-16
ints, 2-20
modular representation, 7-35
unary, 7-8
successor relation, 8-36
internet addresses, 9-22
intersection (of sets), 2-33
intervals, see real numbers
invalid inference, 4-75
inverse
additive, 7-55
multiplicative, 7-44 ff.
of a function, 2-80
of a matrix, 2-68
of a relation, 8-8 ff., 8-29
of an implication, 3-25
IP addresses, 9-22
irrational numbers, see rationals
irrationality of √2, 4-39
irreflexivity, 8-25 ff.
isomorphism (of graphs), 11-16 ff.
Jaccard coefficient, 2-45
Java (programming language), 2-73, 3-12, 3-34, 11-51
Johnson’s algorithm, 10-85
join (database operation), 8-17
K, see complete graphs
K₂, see bipartite graphs
Kasparov, Garry, 3-54
Keller, Mary Kenneth, 4-75
keyspace, see hashing
kidney transplants, 11-71, 12-2
Knut, Donald, 7-12
Kruskal’s algorithm, 11-87
Kuratowski’s Theorem, 11-25
L (complexity class), 6-35
latchstring, 8-40, 11-81
law of the excluded middle, 3-22
Law of Total Expectation, 10-72
Law of Total Probability, 10-43
least common multiple, 7-11 ff.
length (of a vector), 2-52
lexical scope, 3-56
lexicographic ordering, 3-63, 8-7
Liou’s Paradox, 2-31
linearity of expectation, 10-64
linked lists, 5-58
adjacency lists for graphs, 11-12 ff.
as graphs, 11-31
recursive definition, 5-45
list, see sequence
little o and little ω, 6-10 ff., 8-31 ff.
logarithms, 2-10–2-11
asymptotics, 6-9 ff.
discrete logarithm, 7-67
polylogarithmic functions, 6-19, 7-8
logic
Boolean logic, 2-4, 7-46
consistency, 3-58
fuzzy logic, 3-17
incompleteness, 3-58
logical equivalence, 3-24, 3-47
logical fallacy, see fallacy
modal logic, 8-32
predicate logic, 3-40 ff.
games against the demon, 3-69
nested quantifiers, 3-63 ff.
order of quantification, 3-65 ff.
predicates, 3-40 ff.
quantifiers, 3-42 ff.
theorems in predicate logic, 3-47 ff.
propositional logic, 3-4 ff.
atomic vs. compound propositions, 3-6
logical connectives, 3-6 ff.
propositions, 3-4 ff.
recursive definition of a well-formed
formula, 5-47
satisfiability, 3-23
tautology, 3-22 ff.
truth assignment, 3-13
truth tables, 3-13 ff.
truth values, 3-4, 5-48
universal set of operators, 4-72
temporal logic, 8-32
longest common subsequence, 5-17, 9-80
loop invariants, 5-20
Lovelace, Ada, 2-22, 2-48
machine learning
bias, 2-62
classification problems, 9-35, 10-50
clustering, 2-42
cross-validation, 9-79
macros, 3-56
Manhattan distance, 2-52, 2-66, 4-72
map, 2-40
Mapping Rule, 9-34 ff.
MapReduce, 2-40
maps, 4-48, 11-24
mark-and-sweep, 11-51
Markov’s inequality, 10-84
matchings, see graphs
matrices, 2-55 ff.
adjacency matrices for graphs, 11-13 ff.
identity matrix, 2-56
inverse of a matrix, 2-68
matrix multiplication, 2-57 ff.
Strassen’s algorithm, 6-68
rotation matrices, 2-63, 4-73
term–document matrix, 2-61
maximal element, 8-54 ff.
maximum element, 2-35, 2-84, 8-54 ff.
mazes, 11-46
median (of an array), 2-89, 10-78 ff.
memorization, 9-73
memory management, 11-51
Merge Sort, see sorting
metrics, 4-5, 4-25–4-26, 11-54
Milgram, Stanley, 4-46
Miller–Rabin test, 4-68, 7-53
minimal element, 8-54 ff.
minimum element, 2-35, 8-54 ff.
minimum spanning trees, 11-85 ff.
cycle rule, 11-86
Kruskal’s algorithm, 11-87
weighted cycle elimination algorithm,
11-86
ML (programming language), 3-73, 5-52
modal logic, 8-32
modal arithmetic, 2-11–2-13, 7-4 ff.
Division Theorem, 7-4
\texttt{mod-and-div} algorithm, 7-6 ff., 7-18
modular congruences, 7-8
modular exponentiation, 7-19
modular products, 7-9
modular sums, 7-9
multiplicative inverse, 7-44 ff.
primitive roots, 7-67
Modus Ponens, 3-23
Modus Tollens, 3-23
Monte Carlo method, 10-76
Monty Hall Problem, 10-14
Moore’s Law, 6-16
multiples, see divisibility
multiplicative identity, 7-44
multiplicative inverse, 7-44 ff.
multitasking, 6-33
naive Bayes classifier, 10-50
\texttt{nand} (not and), 4-57, 5-60
\texttt{n-ary} relations, 8-14 ff.
expressing \texttt{n-ary} relations as binary
relations, 8-15
natural language processing, 12-3
ambiguity, 3-17
language model, 10-48
speech processing, 2-42, 9-32
speech recognition, 10-48
text classification, 10-50
text-to-speech systems, 9-32
natural logarithm, see logarithms
neighbors (in graphs), 11-7, 11-10
nested quantifiers, 3-63 ff.
games against the demon, 3-69
negations, 3-67
order of quantification, 3-65 ff.
nested sums, 2-16, 10-63
Newton’s method, 2-22
nodes, see graphs
nonconstructive proofs, 4-41
not (not or), 4-58, 5-60
not (\neg), 2-70, 3-7
NP (complexity class), 3-32, 4-79, 6-35
number line, 2-6
text classification, see \texttt{naive}

number theory, see modular arithmetic
numerical methods, see scientific
computing
\texttt{O} (Big \texttt{O}), 6-5 ff., 8-31 ff.
\texttt{o} (little \texttt{o}), 6-10, 8-31 ff.
odd numbers, 2-13
off-by-one error, 11-34
Omega (\texttt{O}) (asymptotics), 6-10, 8-31 ff.
omega (\textit{\omega}) (asymptotics), 6-10, 8-31 ff.
one-time pads, 7-58

This material will be published by Cambridge University Press as Connecting Discrete Mathematics and Computer Science by David Liben-Nowell, and an older edition of the material was published by John Wiley & Sons, Inc as Discrete Mathematics for Computer Science. This pre-publication version is free to view and download for personal use only. Not for re-distribution, re-sale, or use in derivative works. © David Liben-Nowell 2020–2021. This version was posted on September 8, 2021.
Index 12-11

one-to-one functions, 2-78, 8-14, 9-35
onto functions, 2-77, 3-63, 8-14, 9-35
operating systems, 3-75
multitasking, 6-33
virtual memory, 4-70
optimizing compilers, 3-34, 3-37
or (\lor), 3-7
order of operations, see precedence of operators
orders, see partial orders
organ donation, 11-71
out-degree, see degree
out-neighbor, see neighbors (in graphs)
outcome (probability), 10-6
overfitting, 10-48
overflow, 2-20, 4-82
\mathcal{P}, see power set
P (complexity class), 3-32, 4-79, 6-35
PageRank, 11-90
Painter’s Algorithm, 8-61
pairwise independence, 10-36
palindromes, 5-60, 9-50
paradoxes
birthday paradox, 10-68
class-size paradox, 10-61
Liar’s paradox, 2-31
nontransitive dice, 10-82
paradoxes of translation, 3-5
Russell’s paradox, 2-31
Simpson’s Paradox, 4-87
voting paradoxes, 8-38
parallel edges, 11-5
parent (in a tree), 11-62
parity, 2-13, 4-15 ff., 5-29–5-30, 5-40
parsing, 5-56
partial orders, 8-50 ff.
chains and antichains, 8-65
comparability, 8-50
extending to a total order, 8-56 ff.
Hasse diagrams, 8-52
immediate successors, 8-53
minimal/maximal elements, 8-54
minimum/maximum element, 8-54
strict partial order, 8-50
topological ordering, 8-56 ff.
total orders, 8-50
consistency with a partial order, 8-56 ff.
partition (of a set), 2-38
bipartite graphs, 11-21
equivalence relations, 8-47
Pascal’s identity, 9-66, 9-71
Pascal’s triangle, 9-70 ff.
paths (in graphs), 11-34 ff.
breadth-first search, 11-42 ff.
connected graphs, 11-36 ff.
depth-first search, 11-46 ff.
Dijkstra’s algorithm, 11-80 ff.
internet routing, 9-22
shortest paths, 11-41 ff.
simple paths, 11-35
Peirce’s arrow (\lnot \lor), 4-58, 5-60
Pentium chip, 4-82, 6-16
perfect matchings, see graphs
perfect numbers, 2-26
perfect square, 2-9
Perl (programming language), 4-60
permutations, 5-42, 9-16–9-17, 9-27
k-permutations, 9-59 ff.
Petersen graph, 11-17, 11-25
Pigeonhole Principle, 9-43 ff., 9-48
planar graphs, 4-48, 11-23 ff.
Kuratowski’s Theorem, 11-25
polygons, 2-63, 5-27, 5-32, 5-36, 5-42, 10-76
polylogarithmic, 6-19, 7-8
polynomials, 2-81 ff., 4-23, see also P (complexity class)
asymptotics, 6-8 ff.
evaluating modulo a prime, 7-25, 7-36, 7-38
postfix notation, 8-6
Postscript (programming language), 8-6
power set, 2-39
as a relation, 8-6
cardinality, 9-37
power-law distribution, 11-26
powers, see exponents
precedence of operators, 2-34, 3-12, 3-45, 5-56
predicate logic, see logic
predicates, 3-40 ff., 8-7, see also logic
prefix notation, 8-6
prefix-free codes, 9-19
preorder, 8-52
prime numbers, 2-13, 4-63, 7-21 ff.
Carmichael numbers, 7-51, 7-56
distribution of the primes, 7-22
infinite set of primes, 4-60
primality testing, 4-60, 4-68, 6-22, 7-21
efficient algorithms, 7-53
prime factorization, 7-24, 7-66
cryptography, 4-68, 7-65
equivalence of, 5-30–5-32
Shor’s algorithm, 10-22
uniqueness of, 7-28–7-30
Prime Number Theorem, 7-22
Sieve of Eratosthenes, 7-22, 7-40
priority queues, 5-38
privacy, 5-22, 10-20, 12-2
probability
Bayes’ Rule, 10-44 ff.
conditional expectation, 10-71 ff.
conditional probability, 10-36 ff.
coupon collector problem, 10-84
events, 10-8 ff.
expectation, 10-60 ff.
infinite probabilities, 10-40
Law of Total Expectation, 10-72
Law of Total Probability, 10-43
linearity of expectation, 10-64 ff.
Markov’s inequality, 10-84
Monty Hall Problem, 10-14
outcomes, 10-6 ff.
probability functions, 10-6 ff.
random variables, 10-57 ff.
random walks, 11-90
standard deviation, 10-72 ff.
tree diagrams, 10-12 ff.
variance, 10-72 ff.

This material will be published by Cambridge University Press as Connecting Discrete Mathematics and Computer Science by David Liben-Nowell, and an older edition of the material was published by John Wiley & Sons, Inc as Discrete Mathematics for Computer Science. This pre-publication version is free to view and download for personal use only. Not for re-distribution, re-sale, or use in derivative works. © David Liben-Nowell 2020-2021. This version was posted on September 8, 2021.
probability distributions
- Bernoulli, 10-16 ff.
- binomial, 10-17 ff.
- entropy, 10-24
- geometric, 10-18 ff.
- posterior distribution, 10-46
- prior distribution, 10-46
- uniform, 10-16 ff.
- product, 2-18 ff.
- of a set, 2-35
- product of sums, see conjunctive normal form

Product Rule, 9-8
- cardinality of $S^k$, 9-9

programming languages
- compile-time optimization, 3-34
- Currying, 3-73
- garbage collection, 6-33, 11-51
- higher-order functions, 2-40, 3-73
- parsing, 5-56
- scoping/functions/macros, 3-56
- short-circuit evaluation, 3-34
- syntactic sugar, 2-28

project (database operation), 8-17
- proofs, 4-30 ff.
  - by assuming the antecedent, 3-51, 4-33
  - by cases, 4-18, 4-34 ff.
  - by construction, 4-14, 4-41 ff.
  - by contradiction, 4-21, 4-38 ff.
  - by contrapositive, 4-36 ff.
  - by induction, 2-83, 5-4 ff.
  - by mutual implication, 4-37
  - by strong induction, 5-28 ff.
  - by structural induction, 5-48 ff.
  - combinatorial proofs, 9-64 ff.
  - direct, 4-32 ff.
  - nonconstructive, 4-41
  - strategy for proofs, 4-42 ff.
  - unprovable true statements, 3-58
  - “without loss of generality”, 4-35
  - writing proofs, 4-44 ff.
- proper subset and superset, 2-36
- propositional logic, see logic
- proving true, see fallacy
- pseudocode, 2-84
- pseudorandom generator, 10-15
- PSPACE (complexity class), 6-35
- public-key cryptography, see cryptography
- Pythagorean Theorem, 4-44, 4-58–4-60, 4-72
- incorrect published proof, 4-87
- Python (programming language), 2-20, 2-40, 2-73, 3-20, 3-56, 3-73, 4-62 ff., 9-46, 11-51
- \( \mathbb{Q} \), see rationals
- QR codes, 4-4, 4-23
- quadrees, 6-57
- quantifiers, 3-42 ff.
  - negating quantifiers, 3-49 ff.
  - nested quantifiers, 3-63 ff.
- vacuous quantification, 3-52
- quantum computation, 10-22
- Quick Sort, see sorting

\( \mathbb{R} \), see real numbers
- Radix Sort, see sorting
- raising to a power, see exponentials

Random Surfer Model, 11-90
- random variables, 10-57 ff.
  - expectation, 10-60 ff.
  - independent random variables, 10-59
  - indicator random variables, 10-59
- random walks, 11-90, 11-93
- randomized algorithms, 6-32
- Buffon’s needle, 10-76
- finding medians, 10-78
- Johnson’s algorithm, 10-85
- Monte Carlo method, 10-76
- primality testing (Miller–Rabin), 7-53
- Quick Sort, 10-27
- range (of a function), 2-73
- rate (of a code), 4-8 ff.
- ratios, 2-4 ff., 2-50, 4-33, 4-37, 7-11
- in lowest terms, 7-11, 8-47
- real numbers, 2-4 ff.
- absolute value/floor/ceiling, 2-7 ff.
- approximate equality (=), 2-6, 8-4
  - defining via infinite sequences, 8-48
  - exponentiation, 2-8 ff.
- floats (representation), 2-20
- intervals, 2-6
- logarithms, 2-10 ff.
- trichotomy, 6-14
- realization, see outcome (probability)
- recommender system, 2-45
- recurrence relations, 6-42 ff.
- divide and conquer, 6-61 ff.
- iterating, 6-45
- sloppiness, 6-49
- solving by induction, 6-44
- variable substitution, 6-47
- recursion tree, 6-40, 6-61 ff.
- recursively defined structures, 5-45 ff.
- Reed–Solomon codes, 4-23, 7-38
- reference counting, 11-51
- refining equivalence relations, 8-48 ff.
- reflexivity, 4-6, 8-25 ff.
- reflexive closure, 8-33 ff.
- regular expressions, 8-40, 8-59
- regular graphs, 11-22
- reindexing, 2-15

relational databases, see databases
- relations
  - n-ary relations, 8-14 ff.
  - binary relations, 8-5 ff.
  - closures, 8-33 ff.
  - composition, 8-9 ff.
  - equivalence relations, 8-45 ff.
  - functions as relations, 8-12 ff.
  - inverses, 8-8 ff.
  - partial orders, 8-50 ff.
  - reflexivity, 8-25
- relational databases, 8-17
- symmetry, 8-26
- total orders, 8-50 ff.
- transitivity, 8-29 ff.
- visual representation, 8-7 ff., 8-24 ff.
- Hasse diagrams, 8-52 ff.
- vs. predicates, 8-7
- relative primality, 7-24 ff., 7-46 ff.
Chinese Remainder Theorem, 7-30 ff.
Extended Euclidean algorithm, 7-27
remainder, see mod
repeated squaring, 6-58, 7-19, 7-63
repetition code, 4-13 ff.
roots (of a polynomial), 2-82, 4-23, 7-38
rotation matrices, see matrices
roulette, 10-11, 10-73
RSA cryptosystem, 4-68, 5-22, 7-60 ff.
breaking the encryption, 7-65
Rubik’s cube, 7-46, 9-28
running time, 6-22 ff.
average case, 6-29 ff., 10-70
best case, 6-29 ff.
worst case, 6-23 ff.
Russell’s paradox, 2-31
sample space (probability), 10-6
sampling bias, 10-61
satisfiability, 3-23, 3-32, 4-63, 8-4, 10-85
scalars, 2-51
SCC, see strongly connected components
Scheme (programming language), 2-40,
2-50, 3-28, 3-73, 8-6
scientific computing, 6-23
Newton’s method, 2-22
seam carving, 9-73
searching
Binary Search, 5-20, 5-42, 6-28 ff.,
6-44, 6-48–6-49, 6-60, 7-18
Linear Search, 6-27 ff.
Ternary Search, 6-57
secret sharing, 7-36, 9-79
select, see median
select (database operation), 8-17
Selection Sort, see sorting
self-loops, 11-5
self-reference, 2-31, 3-5, 3-58, 4-61,
11-90, 12-13
sentinels, 9-57
sequences, 2-48 ff.
$S^p$ (sequence of elements from the same set), 2-50
cardinality, 9-8, 9-14
sets, 2-28 ff.
cardinality, 2-28 ff., 9-4 ff.
characteristic function, 8-7
complement, 2-33
disjointness, see disjoint sets, see also partitions
empty set, 2-32
intersection, 2-33
set difference, 2-34
singleton set, 2-32
subsets/supersets, 2-35 ff., see also
power set
union, 2-33
inclusion–exclusion, 9-10 ff.
Venn diagrams, 2-32
well-ordered, 5-49
Sheffer stroke (|), 4-57, 5-60
Shor’s algorithm, 10-22
short-circuit evaluation, 3-34
Sierpiński triangle/carpet, 5-25 ff.
Sieve of Eratosthenes, 7-22, 7-40
signed social networks, 11-18
Simpson’s Paradox, 4-87
six degrees of separation, 4-46
skite, 11-56
small-world phenomenon, 4-46, 11-49
social networks, 4-46, 11-2, 11-18, 11-26
Dunbar’s number, 11-30
sorting
Bubble Sort, 6-26, 6-32, 6-37
comparison-based, 6-37, 9-24
Counting Sort, 6-38, 9-27
Insertion Sort, 5-10, 6-25, 6-30, 6-37
average-case analysis, 10-70, 10-83
correctness using loop invariants,
5-20
lower bounds, 9-24–9-27
Merge Sort, 5-42, 6-40–6-43,
6-46–6-47, 6-58, 6-60
Quick Sort, 6-38, 6-58
correctness (for any pivot rule),
5-33 ff.
randomized pivot selection, 10-27
Radix Sort, 6-38
Selection Sort, 6-24, 6-31, 6-37, 9-24
spam filters, 10-50
spawning trees, 11-68 ff.
cycle elimination algorithm, 11-69
minimum spanning trees, 11-85 ff.
speech processing, see natural language
processing
sphere packing, 4-21
spreadsheets, 8-58, 8-65, 11-40
SQL (programming language), 8-17
square roots, 2-9, 2-22, see exponentials
Heron’s method, 2-22, 4-50
standard deviation, 10-72 ff.
stenography, 5-22
Strassen’s algorithm, 6-68
strings, 2-50, 2-65
generating all strings of a given length,
7-16
regular expressions, 8-40
strong induction, see proofs
strongly connected components, 11-38 ff.
structural induction, see proofs
subgraphs, see graphs
subsequences, 5-17, 8-21, 9-51, 9-64,
9-78, 9-80
subset, 2-36
Sudoku, 3-32
sun of products, see conjunctive normal form
Sum Rule, 9-5
summations, 2-14 ff.
arithmetic, 2-15, 5-14, 5-25
geometric, 2-15, 5-12 ff., 6-61 ff.
infinite, 5-14
harmonic, 5-14 ff.
of a set, 2-35
reindexing summations, 2-15
reversing nested summations, 2-17,
10-62
superset, 2-36
surjective functions, see onto functions
symmetry, 4-6, 8-5, 8-26 ff.
symmetric closure, 8-33 ff.
syntactic sugar, 3-28
### Index

<table>
<thead>
<tr>
<th>Term</th>
<th>Pages</th>
</tr>
</thead>
<tbody>
<tr>
<td>syzygy</td>
<td>11-25, 11-81</td>
</tr>
<tr>
<td>( T(n) = aT(n/b) + n^k ), ( k &gt; 0 )</td>
<td>6-61 ff.</td>
</tr>
<tr>
<td>tautology</td>
<td>3-22 ff.</td>
</tr>
<tr>
<td>temporal logic</td>
<td>8-32</td>
</tr>
<tr>
<td>term frequency–inverse document frequency ( (\text{TFIDF}) )</td>
<td>2-61</td>
</tr>
<tr>
<td>The Book (of proofs)</td>
<td>4-46</td>
</tr>
<tr>
<td>Theta (( \Theta )) (asymptotics)</td>
<td>6-10, 8-31 ff.</td>
</tr>
<tr>
<td>tic-tac-toe</td>
<td>3-54, 9-52</td>
</tr>
<tr>
<td>topological ordering</td>
<td>8-56 ff.</td>
</tr>
<tr>
<td>Tor</td>
<td>5-22, 12-2</td>
</tr>
<tr>
<td>total orders</td>
<td>8-50 ff., see also partial orders</td>
</tr>
<tr>
<td>totient function</td>
<td>7-56, 9-30</td>
</tr>
<tr>
<td>Towers of Hanoi</td>
<td>6-70</td>
</tr>
<tr>
<td>transitivity</td>
<td>8-29 ff.</td>
</tr>
<tr>
<td>nontransitive dice</td>
<td>10-82</td>
</tr>
<tr>
<td>nontransitivity in voting</td>
<td>8-38</td>
</tr>
<tr>
<td>signed social networks</td>
<td>11-18</td>
</tr>
<tr>
<td>transitive closure</td>
<td>8-33 ff.</td>
</tr>
<tr>
<td>Traveling Salesperson Problem</td>
<td>9-73</td>
</tr>
<tr>
<td>trees</td>
<td>11-57 ff.</td>
</tr>
<tr>
<td>2–3 and 2–3–4 trees</td>
<td>5-59</td>
</tr>
<tr>
<td>AVL trees</td>
<td>6-53 ff.</td>
</tr>
<tr>
<td>binary search trees</td>
<td>5-58, 6-53, 11-73</td>
</tr>
<tr>
<td>binary trees</td>
<td>5-46, 6-53 ff., 11-65 ff.</td>
</tr>
<tr>
<td>complete binary trees</td>
<td>11-77 ff.</td>
</tr>
<tr>
<td>heaps</td>
<td>2-88, 5-38</td>
</tr>
<tr>
<td>decision trees</td>
<td>9-26</td>
</tr>
<tr>
<td>forests</td>
<td>11-60</td>
</tr>
<tr>
<td>game trees</td>
<td>3-54, 9-52</td>
</tr>
<tr>
<td>in counting problems</td>
<td>9-20</td>
</tr>
<tr>
<td>parse trees</td>
<td>5-56</td>
</tr>
<tr>
<td>quadrees</td>
<td>6-57</td>
</tr>
<tr>
<td>recursion trees</td>
<td>6-40 ff.</td>
</tr>
<tr>
<td>recursive definitions of trees</td>
<td>5-46, 11-65</td>
</tr>
<tr>
<td>rooted trees</td>
<td>11-62 ff.</td>
</tr>
<tr>
<td>spanning trees</td>
<td>11-68 ff.</td>
</tr>
<tr>
<td>minimum spanning trees</td>
<td>11-85 ff.</td>
</tr>
<tr>
<td>subtrees</td>
<td>11-63 ff.</td>
</tr>
<tr>
<td>tree traversal</td>
<td>11-65 ff.</td>
</tr>
<tr>
<td>van Emde Boas trees</td>
<td>6-71</td>
</tr>
<tr>
<td>triangle inequality</td>
<td>4-6, 4-35</td>
</tr>
<tr>
<td>triangulation</td>
<td>5-32–5-33, 5-36</td>
</tr>
<tr>
<td>truth tables</td>
<td>3-13 ff.</td>
</tr>
<tr>
<td>truth values</td>
<td>3-4 ff.</td>
</tr>
<tr>
<td>tsktkska</td>
<td>8-40, 11-81</td>
</tr>
<tr>
<td>tuple, see sequence</td>
<td></td>
</tr>
<tr>
<td>Turing Award</td>
<td>2-30, 3-26, 4-5, 6-5, 7-12, 7-60, 7-67, 8-6, 8-17, 11-80</td>
</tr>
<tr>
<td>Turing machines</td>
<td>3-58, 4-62, 6-23</td>
</tr>
<tr>
<td>Turing, Alan</td>
<td>2-30, 4-61, 6-23, 9-75</td>
</tr>
<tr>
<td>unary numbers, see integers</td>
<td></td>
</tr>
<tr>
<td>uncomputability</td>
<td>3-58, 4-61–4-66, 4-70, 9-46</td>
</tr>
<tr>
<td>undecidability, see uncomputability</td>
<td></td>
</tr>
<tr>
<td>underflow</td>
<td>2-20</td>
</tr>
<tr>
<td>Unicode</td>
<td>9-30</td>
</tr>
<tr>
<td>uniform distribution</td>
<td>10-8, 10-16 ff.</td>
</tr>
<tr>
<td>unigrams</td>
<td>10-48</td>
</tr>
<tr>
<td>union (of sets)</td>
<td>2-33</td>
</tr>
<tr>
<td>Union Bound</td>
<td>9-6</td>
</tr>
<tr>
<td>unit vector</td>
<td>2-53</td>
</tr>
<tr>
<td>universal quantifier ( (\forall) )</td>
<td>3-42 ff.</td>
</tr>
<tr>
<td>unsatisfiability</td>
<td>3-23</td>
</tr>
<tr>
<td>URL squatting</td>
<td>9-52</td>
</tr>
<tr>
<td>vacuous quantification</td>
<td>3-52</td>
</tr>
<tr>
<td>valid inference</td>
<td>4-75</td>
</tr>
<tr>
<td>van Emde Boas trees</td>
<td>6-71</td>
</tr>
<tr>
<td>variance</td>
<td>10-72 ff.</td>
</tr>
<tr>
<td>Vector Space Model</td>
<td>2-61</td>
</tr>
<tr>
<td>vectors</td>
<td>2-51 ff.</td>
</tr>
<tr>
<td>dot product</td>
<td>2-53 ff.</td>
</tr>
<tr>
<td>Venn diagrams</td>
<td>2-32</td>
</tr>
<tr>
<td>virtual memory</td>
<td>4-70</td>
</tr>
<tr>
<td>Von Koch snowflake</td>
<td>5-2, 5-10, 5-25 ff.</td>
</tr>
<tr>
<td>Voronoi diagram</td>
<td>2-66</td>
</tr>
<tr>
<td>voting systems</td>
<td>8-38</td>
</tr>
<tr>
<td>wall clocks</td>
<td>6-33</td>
</tr>
<tr>
<td>Weizenbaum, Joseph</td>
<td>12-3</td>
</tr>
<tr>
<td>well-ordered set</td>
<td>5-49</td>
</tr>
<tr>
<td>What Three Words</td>
<td>9-2</td>
</tr>
<tr>
<td>“without loss of generality”</td>
<td>4-35</td>
</tr>
<tr>
<td>word2vec</td>
<td>2-62</td>
</tr>
<tr>
<td>World War II</td>
<td>9-75, 11-18</td>
</tr>
<tr>
<td>World-Wide Web</td>
<td>11-26, 11-49</td>
</tr>
<tr>
<td>PageRank</td>
<td>11-90</td>
</tr>
<tr>
<td>worst-case analysis, see running time</td>
<td></td>
</tr>
<tr>
<td>xor, see exclusive or</td>
<td></td>
</tr>
<tr>
<td>( \mathbb{Z} ), see integers</td>
<td></td>
</tr>
<tr>
<td>( \mathbb{Z}_n ), 7-43 ff.</td>
<td></td>
</tr>
<tr>
<td>zero (of a binary operator)</td>
<td>3-19, 5-59</td>
</tr>
<tr>
<td>zyzygyvas</td>
<td>8-7</td>
</tr>
</tbody>
</table>
References


12-16 References


[39] Roger Dingledine, Nick Mathewson, and Paul Syverson. Tor: The second-generation onion
References 12-17


References


[87] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. Distributed representations of words and phrases and their compositionality. In *Proceedings of the 26th International Conference on Neural...*


References


