Algorithms Cheat Sheet
Pocket Edition

Marcin Woch

Mathematical Operations

Arithmetics

Operation	Algorithm	Input	Output	Complexity
Addition	Schoolbook	Two n-digit numbers matrices	One $n+1$-digit number	$O(n)$
Subtraction	Schoolbook	Two n-digit numbers matrices	One $n+1$-digit number	$O(n)$
Multiplication	Schoolbook	Two n-digit numbers matrices	One $2n$-digit number	$O(n^2)$
Multiplication	3-way Toom-Cook algorithm	Two n-digit numbers matrices	One $2n$-digit number	$O(n^{\log_3 5}) \approx O(n^{1.465})$
Multiplication	k-way Toom-Cook algorithm	Two n-digit numbers matrices	One $2n$-digit number	$O\left(n^{\frac{\log(2k-1)}{\log k}}\right)$
Multiplication	Mixed-level Toom-Cook algorithm	Two n-digit numbers matrices	One $2n$-digit number	$O(n2^{\sqrt{2\log n}}\log n)$
Multiplication	Karatsuba algorithm	Two n-digit numbers matrices	One $2n$-digit number	$O(n^{\log_2 3}) \approx O(n^{1.585})$
Multiplication	Schönhage–Strassen algorithm	Two n-digit numbers matrices	One $2n$-digit number	$O(n\log n\log\log n)$
Multiplication	Harvey-Hoeven algorithm	Two n-digit numbers matrices	One $2n$-digit number	$O(n\log n)$
Multiplication	Pointer machine [1]	Two n-digit numbers matrices	One $2n$-digit number	$O(n)$
Multiplication	Unit Cost RAM machine [1]	Two n-digit numbers matrices	One $2n$-digit number	$O(n)$
Division	Schoolbook	Two n-digit numbers matrices	One n-digit number	$O(n^2)$
Division	Burnikel–Ziegler Divide-and-Conquer[2]	Two n-digit numbers matrices	One n-digit number	$O(M(n)\log n)$
Division	Newton–Raphson division[2]	Two n-digit numbers matrices	One n-digit number	$O(M(n))$
Square root	Newton's method[2]	One n-digit number	One n-digit number	$O(M(n))$
Modular exponentation	Repeated multiplication and reduction[2]	Two n-digit integers, k-bit exponent	One n-digit integer	$O(M(n)2^k)$
Modular exponentation	Exponentiation by squaring[2]	Two n-digit integers, k-bit exponent	One n-digit integer	$O(M(n)k)$
Modular exponentation	Exponentiation with Montgomery reduction[2]	Two n-digit integers, k-bit exponent	One n-digit integer	$O(M(n)k)$

[1] Theoretical model only

[2] $M(n)$ - The complexity of an implemented multiplication algorithm

Matrix Algebra

Operation	Algorithm	Input	Output	Complexity
Multiplication	Schoolbook	Two $n \times n$ matrices	One $n \times n$ matrix	$O(n^3)$
Multiplication	Strassen's	Two $n \times n$ matrices	One $n \times n$ matrix	$O(n^{\log_2 7}) = O(n^{2.807})$
Multiplication	Coppersmith-Winograd	Two $n \times n$ matrices	One $n \times n$ matrix	$O(n^{2.376})$
Multiplication	Alman-Williams	Two $n \times n$ matrices	One $n \times n$ matrix	$O(n^{2.3728596})$
Multiplication	Duan, Wu, Zhou	Two $n \times n$ matrices	One $n \times n$ matrix	$O(n^{2.3719})$
Multiplication	Williams, Xu, Xu, Zhou	Two $n \times n$ matrices	One $n \times n$ matrix	$O(n^{2.3716})$
Multiplication	Schoolbook	One $n \times m$ matrix, one $m \times p$ matrix	One $n \times p$ matrix	$O(nmp)$
Inversion	Gauss-Jordan elimination	One $n \times n$ matrix	One $n \times n$ matrix	$O(n^3)$
Inversion	Strassen algorithm	One $n \times n$ matrix	One $n \times n$ matrix	$O(n^{2.807})$
Inversion	Coppersmith-Winograd algorithm	One $n \times n$ matrix	One $n \times n$ matrix	$O(n^{2.376})$
Inversion	Optimised CW algorithm	One $n \times n$ matrix	One $n \times n$ matrix	$O(n^{2.373})$
SVD	Bidiagonalization, QR algorithm	One $m \times n$ matrix ($m \leq n$)	One $m \times m$ One $m \times n$ matrix One $n \times n$	$O(m^2 n)$
Determinant	Laplace expansion	One $n \times n$ matrix	One number	$O(n!)$
Determinant	Division free algorithm	One $n \times n$ matrix	One number	$O(n^4)$
Determinant	LU decomposition	One $n \times n$ matrix	One number	$O(n^3)$
Determinant	Bareiss algorithm	One $n \times n$ matrix	One number	$O(n^3)$
Determinant	Fast matrix multiplication	One $n \times n$ matrix	One number	$O(n^{2.373})$
Back substitution	Back substitution algorithm	Triangular matrix	n solutions	$O(n^2)$

Polynomials

Operation	Algorithm	Input	Output	Complexity
Polynomial evaluation	Direct	One polynomial of degree n and integer coefficients	One number	$O(n)$
Polynomial evaluation	Horner's algorithm	One polynomial of degree n and integer coefficients	One number	$O(n)$
Polynomial gcd	Euclid's algorithm	Two polynomials of degree n and integer coefficients	One number	$O(n^2)$
Polynomial gcd	Lehmer's algorithm (Fast Euclidean) [3]	Two polynomials of degree n and integer coefficients	One number	$O(M(n) \log n)$

[3] $M(n)$ - The complexity of an implemented multiplication algorithm

Number theory

Operation	Algorithm	Input	Output	Complexity
Greatest common divisor	Euclidean algorithm	Two n-digit integers	One integer	$O(n^2)$
Greatest common divisor	Binary GCD	Two n-digit integers	One integer	$O(n^2)$
Greatest common divisor	Left/right k-ary binary GCD	Two n-digit integers	One integer	$O(\frac{n^2}{\log n})$
Greatest common divisor	Stehle-Zimmermann algorithm[4]	Two n-digit integers	One integer	$O(M(n)\log n)$
Greatest common divisor	Schönhage algorithm[4]	Two n-digit integers	One integer	$O(M(n)\log n)$
Jacobi symbol	Stehle-Zimmermann algorithm[4]	Two n-digit integers	0, -1 or 1	$O(M(n)\log n)$
Jacobi symbol	Schönhage algorithm[4]	Two n-digit integers	0, -1 or 1	$O(M(n)\log n)$
Factorial	Bottom-up multiplication[4]	One positive integer less than n	One integer	$O(M(n^2)\log n)$
Factorial	Binary splitting[4]	One positive integer less than n	One integer	$O(M(n\log n)\log n)$
Factorial	Exponentiation of the prime factors of n[4]	One positive integer less than n	One integer	$O(M(n\log n)\log\log n)$
Factorial	Exponentiation of the prime factors of n[4]	One positive integer less than n	One integer	$O(M(n\log n))$
Primality test	AKS primality test n	n digit integer	True or false	$O(n^{6+O(1)})$
Primality test	AKS primality test with Agrawal's conjecture$n$	n digit integer	True or false	$O(n^3)$
Primality test	Elliptic curve test[5] - heuristical approach	n digit integer	True or false	$O(n^{4+\epsilon})$
Primality test	Baillie-PSW test[5]	n digit integer	True or false	$O(n^{2+\epsilon})$
Primality test	Miller-Rabin test[5] [6]	n digit integer	True or false	$O(kn^{2+\epsilon})$
Primality test	Solovay-Strassen test[5] [6]	n digit integer	True or false	$O(kn^{2+\epsilon})$
Integer factorisation	General number field sieve[5]	b-bit input integer	A set of factors	$O((1+\epsilon)^b)$
Integer factorisation	Shor's algorithm [4] [7]	b-bit input integer	A set of factors	$O(M(b)b)$

[4] $M(n)$ - The complexity of an implemented multiplication algorithm
[5] ϵ - a positive constant
[6] k - a positive constant
[7] Theoretical model, on quantum computer

Additional Operation

Special Functions - Elementary functions

Operation	Algorithm	Input	Output	Complexity
Discrete Fourier transform	Schoolbook	Size n data sequence	Set of complex number	$O(n^2)$
Discrete Fourier transform	Fast Fourier transform	Size n data sequence	Set of complex number	$O(n \log n)$
Golden ration	Newton's method[8]			$O(M(n))$
Square root of 2	Newton's method[8]			$O(M(n))$
Euler's number	Taylor series binary splitting of the exp. function[8]			$O(M(n) \log n)$
Euler's number	Newton inversion of the natural logarithm[8]			$O(M(n) \log n)$
Pi	Arctan series binary splitting in Machin's formula[8]			$O(M(n) \log^2 n)$
Pi	Gauss-Legendre algorithm[8]			$O(M(n) \log n)$
Euler's constant	Sweeney's method[8]			$O(M(n) \log^2 n)$
Gamma function	Approx. of the incomplete gamma function[8]	n digit number		$O(M(n)n^{\frac{1}{2}} \log^2 n)$
Gamma function	Hypergeometric series [8]	Fixed ration number		$O(M(n) \log^2 n)$
Hypergeometric function	Borwein and Borwein [8]	n-digit number		$O(M(n)n^{\frac{1}{2}} \log^2 n)$
Hypergeometric function	Hypergeometric series [8]	Fixed rational number		$O(M(n) \log^2 n)$
Taylor series	Repeated argument reduction[8] [9]			$O(M(n)n^{\frac{1}{2}})$
Taylor series	FFT-based acceleration [8] [9]			$O(M(n)n^{\frac{1}{3}} \log^2 n)$
Taylor series	Binary splitting + bit-burst[8] [9]			$O(M(n) \log^2 n)$
Arithmetic-geometric mean iteration	Arithmetic-geometric mean iteration[8] [9]			$O(M(n) \log n)$

[8] $M(n)$ - The complexity of an implemented multiplication algorithm
[9] Aplicability: exp. log, sin, cos, arctan

Sorting Algorithms

Name	Average	Worst	Stable
Quicksort	$n \log n$	n^2	No
Merge sort	$n \log n$	$n \log n$	Yes
In-place merge sort	-	$n \log^2 n$	Yes
Introsort	$n \log n$	$n \log n$	No
Heapsort	$n \log n$	$n \log n$	No
Insertion sort	n^2	n^2	Yes
Block sort	$n \log n$	$n \log n$	Yes
Timsort	$n \log n$	$n \log n$	Yes
Selection sort	n^2	n^2	No
Cubesort	$n \log)$	$n \log n$	Yes
Shellsort	$n^{\frac{4}{3}}$	$n^{\frac{3}{2}}$	No
Bubble sort	n^2	n^2	Yes
Exchange sort	n^2	n^2	No
Tree sort	$n \log n$	$n \log n$	Yes
Cycle sort	n^2	n^2	No
Library sort	$n \log)$	n^2	No
Patience sort	$n \log n$	$n \log n$	No
Smoothsort	$n \log n$	$n \log n$	No
Strand sort	n^2	n^2	Yes
Tournament sort	$n \log n$	$n \log n$	No
Cocktail shaker sort	n^2	n^2	Yes
Comb sort	n^2	n^2	No
Gnome sort	n^2	n^2	Yes
Odd-even sort	n^2	n^2	Yes

Graphs

Operation	Algorithm	Input	Comment	Complexity
Explicit Graph Search	Depth First Search	V - vertices, E - edges		$O(\|V\| + \|E\|)$
Implicit Graph Search	Depth First Search	b - branching factor, d - depth		$O(b^d)$
Explicit Graph Search	Breadth First Search	V - vertices, E - edges		$O(\|V\| + \|E\|)$
Implicit Graph Search	Breadth First Search	b - branching factor, d - depth		$O(b^d)$
Shortest Path	Dijkstra's Algorithm	V - vertices, E - edges	Priority queue/heap	$O(\|V\| + \|E\|)log\|V\|$
Shortest Path	Dijkstra's Algorithm	V - vertices, E - edges	Array	$O(\|V\|^2)$