Algorithms Cheat Sheet
Pocket Edition

Marcin Woch

Mathematical Operations

Arithmetics

Operation	Name	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	$O(n^{\log_3 5}) \approx O(n^1$
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	$O(n2^{\sqrt{2\log n}}\log$
Multiplication	Karatsuba algorithm	Two n-digit numbers matrices	One $2n$-digit number	$O(n^{\log_2 3}) \approx O(n^1$
Multiplication	Schönhage–Strassen algorithm	Two n-digit numbers matrices	One $2n$-digit number	$O(n \log n \log \log$
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 Division[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] The complexity of an implemented multiplication algorithm

Matrix Algebra

Operation	Name	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	Alman-Williams	Two $n \times n$ matrices	One $n \times n$ matrix	$O(n^{2.3728596})$
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	$O(n^2)$

Graphs

Operation	Name	Input	Comment	Bound
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	$O(\|V\|^2)$

Sorting

Name	Average	Worst	Stable
Quicksort	$n \log n$	n^2	No
Merge sort	$n \log n$	$n \log)$	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