

# Algorithms Cheat Sheet

## Pocket Edition

Marcin Woch

# Contents

<b>Introduction</b>	<b>3</b>
<b>Mathematical Operations</b>	<b>4</b>
Arithmetics . . . . .	4
Matrix Algebra . . . . .	5
Polynomials . . . . .	6
Number theory . . . . .	7
Additional Operations . . . . .	8
<b>Data Structures - Common Operations</b>	<b>9</b>
<b>Sorting Algorithms</b>	<b>10</b>
<b>Graphs</b>	<b>11</b>
Data Structure Operations . . . . .	11
Graph Search Algorithms . . . . .	12
Other Graph Algorithms . . . . .	13
<b>Operations on Strings</b>	<b>14</b>

# Introduction

# 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 <sup>1</sup>	Two $n$ -digit numbers matrices	One $2n$ -digit number	$O(n)$
Multiplication	Unit Cost RAM machine <sup>1</sup>	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 <sup>2</sup>	Two $n$ -digit numbers matrices	One $n$ -digit number	$O(M(n) \log n)$
Division	Newton–Raphson division <sup>2</sup>	Two $n$ -digit numbers matrices	One $n$ -digit number	$O(M(n))$
Square root	Newton's method <sup>2</sup>	One $n$ -digit number	One $n$ -digit number	$O(M(n))$
Modular exponentiation	Repeated multiplication and reduction <sup>2</sup>	Two $n$ -digit integers, $k$ -bit exponent	One $n$ -digit integer	$O(M(n)2^k)$
Modular exponentiation	Exponentiation by squaring <sup>2</sup>	Two $n$ -digit integers, $k$ -bit exponent	One $n$ -digit integer	$O(M(n)k)$
Modular exponentiation	Exponentiation with Montgomery reduction <sup>2</sup>	Two $n$ -digit integers, $k$ -bit exponent	One $n$ -digit integer	$O(M(n)k)$

<sup>1</sup> Theoretical model only

<sup>2</sup>  $M(n)$  - 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 $m \times m$	$O(m^2 n)$
			One $m \times n$ matrix	
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) <sup>3</sup>	Two polynomials of degree $n$ and integer coefficients	One number	$O(M(n) \log n)$

---

<sup>3</sup>  $M(n)$  - 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 <sup>4</sup>	Two n-digit integers	One integer	$O(M(n) \log n)$
Greatest common divisor	Schönhage algorithm <sup>4</sup>	Two n-digit integers	One integer	$O(M(n) \log n)$
Jacobi symbol	Stehle-Zimmermann algorithm <sup>4</sup>	Two n-digit integers	0, -1 or 1	$O(M(n) \log n)$
Jacobi symbol	Schönhage algorithm <sup>4</sup>	Two n-digit integers	0, -1 or 1	$O(M(n) \log n)$
Factorial	Bottom-up multiplication <sup>4</sup>	One positive integer less than $n$	One integer	$O(M(n^2) \log n)$
Factorial	Binary splitting <sup>4</sup>	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 conjecturen	$n$ digit integer	True or false	$O(n^3)$
Primality test	Elliptic curve test <sup>5</sup> - heuristical approach	$n$ digit integer	True or false	$O(n^{4+\epsilon})$
Primality test	Baillie-PSW test <sup>5</sup>	$n$ digit integer	True or false	$O(n^{2+\epsilon})$
Primality test	Miller-Rabin test <sup>5</sup> <sup>6</sup>	$n$ digit integer	True or false	$O(kn^{2+\epsilon})$
Primality test	Solovay-Strassen test <sup>5</sup> <sup>6</sup>	$n$ digit integer	True or false	$O(kn^{2+\epsilon})$
Integer factorisation	General number field sieve <sup>5</sup>	$b$ -bit input integer	A set of factors	$O((1 + \epsilon)^b)$
Integer factorisation	Shor's algorithm <sup>4</sup> <sup>7</sup>	$b$ -bit input integer	A set of factors	$O(M(b)b)$

<sup>4</sup>  $M(n)$  - The complexity of an implemented multiplication algorithm

<sup>5</sup>  $\epsilon$  - a positive constant

<sup>6</sup>  $k$  - a positive constant

<sup>7</sup> Theoretical model, on quantum computer

# Additional Operations

## 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 <sup>8</sup>			$O(M(n))$
Square root of 2	Newton's method <sup>8</sup>			$O(M(n))$
Euler's number	Taylor series binary splitting of the exp. function <sup>8</sup>			$O(M(n) \log n)$
Euler's number	Newton inversion of the natural logarithm <sup>8</sup>			$O(M(n) \log n)$
Pi	Arctan series binary splitting in Machin's formula <sup>8</sup>			$O(M(n) \log^2 n)$
Pi	Gauss-Legendre algorithm <sup>8</sup>			$O(M(n) \log n)$
Euler's constant	Sweeney's method <sup>8</sup>			$O(M(n) \log^2 n)$
Gamma function	Approx. of the incomplete gamma function <sup>8</sup>	$n$ digit number		$O(M(n)n^{\frac{1}{2}} \log^2 n)$
Gamma function	Hypergeometric series <sup>8</sup>	Fixed ration number		$O(M(n) \log^2 n)$
Hypergeometric function	Borwein and Borwein <sup>8</sup>	$n$ -digit number		$O(M(n)n^{\frac{1}{2}} \log^2 n)$
Hypergeometric function	Hypergeometric series <sup>8</sup>	Fixed rational number		$O(M(n) \log^2 n)$
Taylor series	Repeated argument reduction <sup>8 9</sup>			$O(M(n)n^{\frac{1}{2}})$
Taylor series	FFT-based acceleration <sup>8 9</sup>			$O(M(n)n^{\frac{1}{3}} \log^2 n)$
Taylor series	Binary splitting + bit-burst <sup>8 9</sup>			$O(M(n) \log^2 n)$
Arithmetic-geometric mean iteration	Arithmetic-geometric mean iteration <sup>8 9</sup>			$O(M(n) \log n)$

<sup>8</sup>  $M(n)$  - The complexity of an implemented multiplication algorithm

<sup>9</sup> Applicability: exp. log, sin, cos, arctan



# Data Structures - Common Operations

# 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	$n \log n$	$n \log n$	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	$n \log n$	$n^2$	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

## Data Structure Operations

# Graph Search Algorithms

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	$O( V ^2)$

# Other Graph Algorithms

# Operations on Strings