Algorithms

**Analysis of Algorithms**: [Asymptotic Analysis](http://www.geeksforgeeks.org/analysis-of-algorithms-set-1-asymptotic-analysis/),  [Worst, Average and Best Cases](http://www.geeksforgeeks.org/analysis-of-algorithms-set-2-asymptotic-analysis/),  [Asymptotic Notations](http://www.geeksforgeeks.org/analysis-of-algorithms-set-3asymptotic-notations/),  [Analysis of Loops](http://www.geeksforgeeks.org/analysis-algorithms-set-4-analysis-programs/),  [Solving Recurrences](http://www.geeksforgeeks.org/analysis-algorithm-set-4-master-method-solving-recurrences/),  [Amortized Analysis](http://www.geeksforgeeks.org/analysis-algorithm-set-5-amortized-analysis-introduction/), [What does ‘Space Complexity’ mean?,](http://www.geeksforgeeks.org/g-fact-86/)[NP-Completeness Introduction](http://www.geeksforgeeks.org/np-completeness-set-1/),  [A Time Complexity Question](http://www.geeksforgeeks.org/a-time-complexity-question/),  [Time Complexity of building a heap](http://www.geeksforgeeks.org/g-fact-85/), [Quiz on Analysis of Algorithms](http://geeksquiz.com/algorithms/analysis-of-algorithms/), [Quiz on Recurrences](http://geeksquiz.com/algorithms/analysis-of-algorithms-recurrences/)

**Searching and Sorting**: [Binary Search](http://geeksquiz.com/binary-search/), [Selection Sort](http://geeksquiz.com/selection-sort/),  [Bubble Sort](http://geeksquiz.com/bubble-sort/), [Insertion Sort](http://geeksquiz.com/insertion-sort/),  [Merge Sort](http://geeksquiz.com/merge-sort/),  [Heap Sort](http://geeksquiz.com/heap-sort/),  [QuickSort](http://geeksquiz.com/quick-sort/" \t "_blank), [Bucket Sort](http://www.geeksforgeeks.org/bucket-sort-2/), [ShellSort](http://geeksquiz.com/shellsort/" \o "Permanent link to ShellSort), [Interpolation search vs Binary search,](http://www.geeksforgeeks.org/g-fact-84/)[Stability in sorting algorithms](http://www.geeksforgeeks.org/stability-in-sorting-algorithms/),  [When does the worst case of Quicksort occur?](http://www.geeksforgeeks.org/when-does-the-worst-case-of-quicksort-occur/),  [Lower bound for comparison based sorting algorithms](http://www.geeksforgeeks.org/lower-bound-on-comparison-based-sorting-algorithms/).  [Which sorting algorithm makes minimum number of memory writes?,](http://www.geeksforgeeks.org/which-sorting-algorithm-makes-minimum-number-of-writes/)[Find the Minimum length Unsorted Subarray, sorting which makes the complete array sorted,](http://www.geeksforgeeks.org/minimum-length-unsorted-subarray-sorting-which-makes-the-complete-array-sorted/)[Merge Sort for Linked Lists](http://www.geeksforgeeks.org/merge-sort-for-linked-list/), [Sort a nearly sorted (or K sorted) array,](http://www.geeksforgeeks.org/nearly-sorted-algorithm/)[Iterative Quick Sort](http://www.geeksforgeeks.org/iterative-quick-sort/), [QuickSort on Singly Linked List](http://www.geeksforgeeks.org/quicksort-on-singly-linked-list/" \t "_blank), [QuickSort on Doubly Linked Lis](http://www.geeksforgeeks.org/quicksort-for-linked-list/" \t "_blank)t, [Find k closest elements to a given value](http://www.geeksforgeeks.org/find-k-closest-elements-given-value/), [Sort n numbers in range from 0 to n^2 – 1 in linear time](http://www.geeksforgeeks.org/sort-n-numbers-range-0-n2-1-linear-time/),  [A Problem in Many Binary Search Implementations](http://www.geeksforgeeks.org/problem-binary-search-implementations/), [Search in an almost sorted array](http://www.geeksforgeeks.org/search-almost-sorted-array/), [Quiz on Sorting](http://geeksquiz.com/algorithms/searching-and-sorting/), [Quiz on Searching](http://geeksquiz.com/algorithms/searching/)

**Greedy Algorithms**: [Activity Selection Problem](http://www.geeksforgeeks.org/greedy-algorithms-set-1-activity-selection-problem/), [Kruskal’s Minimum Spanning Tree Algorithm](http://www.geeksforgeeks.org/greedy-algorithms-set-2-kruskals-minimum-spanning-tree-mst/),[Huffman Coding](http://www.geeksforgeeks.org/greedy-algorithms-set-3-huffman-coding/), [Efficient Huffman Coding for Sorted Input](http://www.geeksforgeeks.org/greedy-algorithms-set-3-huffman-coding-set-2/), [Prim’s Minimum Spanning Tree Algorithm](http://www.geeksforgeeks.org/greedy-algorithms-set-5-prims-minimum-spanning-tree-mst-2/), [Prim’s MST for Adjacency List Representation](http://www.geeksforgeeks.org/greedy-algorithms-set-5-prims-mst-for-adjacency-list-representation/), [Dijkstra’s Shortest Path Algorithm](http://www.geeksforgeeks.org/greedy-algorithms-set-6-dijkstras-shortest-path-algorithm/)

**Dynamic Programming**: [Overlapping Subproblems Property](http://www.geeksforgeeks.org/dynamic-programming-set-1/), [Optimal Substructure Property](http://www.geeksforgeeks.org/dynamic-programming-set-2-optimal-substructure-property/),[Longest Increasing Subsequence](http://www.geeksforgeeks.org/dynamic-programming-set-3-longest-increasing-subsequence/), [Longest Common Subsequence](http://www.geeksforgeeks.org/dynamic-programming-set-4-longest-common-subsequence/), [Edit Distance](http://www.geeksforgeeks.org/dynamic-programming-set-5-edit-distance/), [Min Cost Path](http://www.geeksforgeeks.org/dynamic-programming-set-6-min-cost-path/), [Coin Change](http://www.geeksforgeeks.org/dynamic-programming-set-7-coin-change/), [Matrix Chain Multiplication](http://www.geeksforgeeks.org/dynamic-programming-set-8-matrix-chain-multiplication/), [Binomial Coefficient](http://www.geeksforgeeks.org/dynamic-programming-set-9-binomial-coefficient/), [0-1 Knapsack Problem](http://www.geeksforgeeks.org/dynamic-programming-set-10-0-1-knapsack-problem/),[Egg Dropping Puzzle](http://www.geeksforgeeks.org/dynamic-programming-set-11-egg-dropping-puzzle/), [Longest Palindromic Subsequence](http://www.geeksforgeeks.org/dynamic-programming-set-12-longest-palindromic-subsequence/), [Cutting a Rod](http://www.geeksforgeeks.org/dynamic-programming-set-13-cutting-a-rod/), [Maximum Sum Increasing Subsequence](http://www.geeksforgeeks.org/dynamic-programming-set-14-maximum-sum-increasing-subsequence/), [Longest Bitonic Subsequence](http://www.geeksforgeeks.org/dynamic-programming-set-15-longest-bitonic-subsequence/), [Floyd Warshall Algorithm](http://www.geeksforgeeks.org/dynamic-programming-set-16-floyd-warshall-algorithm/), [Palindrome Partitioning](http://www.geeksforgeeks.org/dynamic-programming-set-17-palindrome-partitioning/), [Partition problem](http://www.geeksforgeeks.org/dynamic-programming-set-18-partition-problem/), [Word Wrap Problem](http://www.geeksforgeeks.org/dynamic-programming-set-18-word-wrap/), [Maximum Length Chain of Pairs](http://www.geeksforgeeks.org/dynamic-programming-set-20-maximum-length-chain-of-pairs/), [Variations of LIS](http://www.geeksforgeeks.org/dynamic-programming-set-14-variations-of-lis/), [Box Stacking Problem](http://www.geeksforgeeks.org/dynamic-programming-set-21-box-stacking-problem/), [Program for Fibonacci numbers](http://www.geeksforgeeks.org/program-for-nth-fibonacci-number/), [Minimum number of jumps to reach end](http://www.geeksforgeeks.org/minimum-number-of-jumps-to-reach-end-of-a-given-array/), [Maximum size square sub-matrix with all 1s](http://www.geeksforgeeks.org/maximum-size-sub-matrix-with-all-1s-in-a-binary-matrix/), [Ugly Numbers](http://www.geeksforgeeks.org/ugly-numbers/), [Largest Sum Contiguous Subarray](http://www.geeksforgeeks.org/largest-sum-contiguous-subarray/), [Longest Palindromic Substring](http://www.geeksforgeeks.org/longest-palindrome-substring-set-1/), [Bellman–Ford Algorithm for Shortest Paths](http://www.geeksforgeeks.org/dynamic-programming-set-23-bellman-ford-algorithm/), [Optimal Binary Search Tree](http://www.geeksforgeeks.org/dynamic-programming-set-24-optimal-binary-search-tree/), [Largest Independent Set Problem](http://www.geeksforgeeks.org/largest-independent-set-problem/), [Subset Sum Problem](http://www.geeksforgeeks.org/dynamic-programming-subset-sum-problem/), [Maximum sum rectangle in a 2D matrix,](http://www.geeksforgeeks.org/dynamic-programming-set-27-max-sum-rectangle-in-a-2d-matrix/)[Count number of binary strings without consecutive 1?s](http://www.geeksforgeeks.org/count-number-binary-strings-without-consecutive-1s/),  [Boolean Parenthesization Problem,](http://www.geeksforgeeks.org/dynamic-programming-set-37-boolean-parenthesization-problem/)[Count ways to reach the n’th stair.](http://www.geeksforgeeks.org/count-ways-reach-nth-stair/)  See [Dynamic Programming Tag](http://www.geeksforgeeks.org/tag/dynamic-programming/)for more problems, [Quiz on Dynamic Programming](http://geeksquiz.com/algorithms/dynamic-programming/)

**Pattern Searching:**[Naive Pattern Searching](http://www.geeksforgeeks.org/searching-for-patterns-set-1-naive-pattern-searching/), [KMP Algorithm](http://www.geeksforgeeks.org/searching-for-patterns-set-2-kmp-algorithm/), [Rabin-Karp Algorithm](http://www.geeksforgeeks.org/searching-for-patterns-set-3-rabin-karp-algorithm/), [A Naive Pattern Searching Question](http://www.geeksforgeeks.org/pattern-searching-set-4-a-naive-string-matching-algo-question/), [Finite Automata](http://www.geeksforgeeks.org/searching-for-patterns-set-5-finite-automata/), [Efficient Construction of Finite Automata](http://www.geeksforgeeks.org/pattern-searching-set-5-efficient-constructtion-of-finite-automata/), [Boyer Moore Algorithm – Bad Character Heuristic](http://www.geeksforgeeks.org/pattern-searching-set-7-boyer-moore-algorithm-bad-character-heuristic/), [Suffix Array](http://www.geeksforgeeks.org/suffix-array-set-1-introduction/), [Anagram Substring Search (Or Search for all permutations)](http://www.geeksforgeeks.org/anagram-substring-search-search-permutations/)

**Backtracking**:  [Print all permutations of a given string](http://www.geeksforgeeks.org/write-a-c-program-to-print-all-permutations-of-a-given-string/), [The Knight’s tour problem](http://www.geeksforgeeks.org/backtracking-set-1-the-knights-tour-problem/), [Rat in a Maze](http://www.geeksforgeeks.org/backttracking-set-2-rat-in-a-maze/),[N Queen Problem](http://www.geeksforgeeks.org/backtracking-set-3-n-queen-problem/), [Subset Sum](http://www.geeksforgeeks.org/backttracking-set-4-subset-sum/), [m Coloring Problem](http://www.geeksforgeeks.org/backttracking-set-5-m-coloring-problem/), [Hamiltonian Cycle](http://www.geeksforgeeks.org/backtracking-set-7-hamiltonian-cycle/), [Sudoku](http://www.geeksforgeeks.org/backtracking-set-7-suduku/),  [Tug of War,](http://www.geeksforgeeks.org/tug-of-war/)[Solving Cryptarithmetic Puzzles](http://www.geeksforgeeks.org/backtracking-set-8-solving-cryptarithmetic-puzzles/)

**Divide and Conquer**:  [Introduction](http://www.geeksforgeeks.org/divide-and-conquer-set-1-find-closest-pair-of-points/), [Write your own pow(x, n) to calculate x\*n](http://www.geeksforgeeks.org/write-a-c-program-to-calculate-powxn/), [Median of two sorted arrays](http://www.geeksforgeeks.org/median-of-two-sorted-arrays/), [Count Inversions](http://www.geeksforgeeks.org/counting-inversions/), [Closest Pair of Points,](http://www.geeksforgeeks.org/closest-pair-of-points/)[Strassen’s Matrix Multiplication,](http://www.geeksforgeeks.org/strassens-matrix-multiplication/)See[this](http://www.geeksforgeeks.org/archives/tag/divide-and-conquer)for more, [Quiz on Divide and Conquer](http://geeksquiz.com/algorithms/divide-and-conquer/)

**Geometric Algorithms:** [Closest Pair of Points | O(nlogn) Implementation](http://www.geeksforgeeks.org/closest-pair-of-points-onlogn-implementation/), [How to check if two given line segments intersect?](http://www.geeksforgeeks.org/check-if-two-given-line-segments-intersect/), [How to check if a given point lies inside or outside a polygon?](http://www.geeksforgeeks.org/how-to-check-if-a-given-point-lies-inside-a-polygon/), [Convex Hull | Set 1 (Jarvis’s Algorithm or Wrapping)](http://www.geeksforgeeks.org/convex-hull-set-1-jarviss-algorithm-or-wrapping/), [Convex Hull | Set 2 (Graham Scan)](http://www.geeksforgeeks.org/convex-hull-set-2-graham-scan/), [Given n line segments, find if any two segments intersect,](http://www.geeksforgeeks.org/given-a-set-of-line-segments-find-if-any-two-segments-intersect/)[Check whether a given point lies inside a triangle or not](http://www.geeksforgeeks.org/check-whether-a-given-point-lies-inside-a-triangle-or-not/)

**Mathematical Algorithms:**[Write an Efficient Method to Check if a Number is Multiple of 3](http://www.geeksforgeeks.org/write-an-efficient-method-to-check-if-a-number-is-multiple-of-3/),  [Efficient way to multiply with 7](http://www.geeksforgeeks.org/efficient-way-to-multiply-with-7/),   [Write a C program to print all permutations of a given string](http://www.geeksforgeeks.org/write-a-c-program-to-print-all-permutations-of-a-given-string/),  [Lucky Numbers](http://www.geeksforgeeks.org/lucky-numbers/),   [Write a program to add two numbers in base 14](http://www.geeksforgeeks.org/write-a-program-to-add-two-numbers-in-base-14/),   [Babylonian method for square root](http://www.geeksforgeeks.org/square-root-of-a-perfect-square/),   [Multiply two integers without using multiplication, division and bitwise operators, and no loops](http://www.geeksforgeeks.org/multiply-two-numbers-without-using-multiply-division-bitwise-operators-and-no-loops/),   [Print all combinations of points that can compose a given number](http://www.geeksforgeeks.org/print-all-combinations-of-points-that-can-compose-a-given-number/),   [Write you own Power without using multiplication(\*) and division(/) operators](http://www.geeksforgeeks.org/write-you-own-power-without-using-multiplication-and-division/),   [Program for Fibonacci numbers](http://www.geeksforgeeks.org/program-for-nth-fibonacci-number/),   [Average of a stream of numbers](http://www.geeksforgeeks.org/average-of-a-stream-of-numbers/),   [Count numbers that don’t contain 3](http://www.geeksforgeeks.org/count-numbers-that-dont-contain-3/),   [Magic Square](http://www.geeksforgeeks.org/magic-square/),   [Sieve of Eratosthenes](http://www.geeksforgeeks.org/sieve-of-eratosthenes/),   [Find day of the week for a given date](http://www.geeksforgeeks.org/find-day-of-the-week-for-a-given-date/),   [DFA based division](http://www.geeksforgeeks.org/dfa-based-division/),  [Generate integer from 1 to 7 with equal probability](http://www.geeksforgeeks.org/generate-integer-from-1-to-7-with-equal-probability/),   [Given a number, find the next smallest palindrome](http://www.geeksforgeeks.org/given-a-number-find-next-smallest-palindrome-larger-than-this-number/),   [Make a fair coin from a biased coin](http://www.geeksforgeeks.org/print-0-and-1-with-50-probability/),   [Check divisibility by 7](http://www.geeksforgeeks.org/divisibility-by-7/),   [Find the largest multiple of 3](http://www.geeksforgeeks.org/find-the-largest-number-multiple-of-3/),   [Lexicographic rank of a string](http://www.geeksforgeeks.org/lexicographic-rank-of-a-string/),   [Print all permutations in sorted (lexicographic) order](http://www.geeksforgeeks.org/lexicographic-permutations-of-string/),   [Shuffle a given array](http://www.geeksforgeeks.org/shuffle-a-given-array/),   [Space and time efficient Binomial Coefficient](http://www.geeksforgeeks.org/space-and-time-efficient-binomial-coefficient/),   [Reservoir Sampling](http://www.geeksforgeeks.org/reservoir-sampling/),   [Pascal’s Triangle](http://www.geeksforgeeks.org/pascal-triangle/),   [Select a random number from stream, with O(1) space](http://www.geeksforgeeks.org/select-a-random-number-from-stream-with-o1-space/),   [Find the largest multiple of 2, 3 and 5](http://www.geeksforgeeks.org/find-the-largest-multiple-of-2-3-and-5/),   [Efficient program to calculate e^x](http://www.geeksforgeeks.org/program-to-efficiently-calculate-ex/),   [Measure one litre using two vessels and infinite water supply](http://www.geeksforgeeks.org/measure-1-litre-from-two-vessels-infinite-water-supply/),   [Efficient program to print all prime factors of a given number](http://www.geeksforgeeks.org/print-all-prime-factors-of-a-given-number/),   [Print all possible combinations of r elements in a given array of size n](http://www.geeksforgeeks.org/print-all-possible-combinations-of-r-elements-in-a-given-array-of-size-n/),   [Random number generator in arbitrary probability distribution fashion](http://www.geeksforgeeks.org/random-number-generator-in-arbitrary-probability-distribution-fashion/),   [How to check if a given number is Fibonacci number?](http://www.geeksforgeeks.org/check-number-fibonacci-number/),   [Russian Peasant Multiplication](http://www.geeksforgeeks.org/fast-multiplication-method-without-using-multiplication-operator-russian-peasants-algorithm/),  [Count all possible groups of size 2 or 3 that have sum as multiple of 3](http://www.geeksforgeeks.org/count-possible-groups-size-2-3-sum-multiple-3/), [Tower of Hanoi](http://geeksquiz.com/c-program-for-tower-of-hanoi/), [Horner’s Method for Polynomial Evaluation](http://www.geeksforgeeks.org/horners-method-polynomial-evaluation/), [Count trailing zeroes in factorial of a number](http://www.geeksforgeeks.org/count-trailing-zeroes-factorial-number/), [Program for nth Catalan Number](http://www.geeksforgeeks.org/program-nth-catalan-number/),  [Generate one of 3 numbers according to given probabilities](http://www.geeksforgeeks.org/write-a-function-to-generate-3-numbers-according-to-given-probabilities/), [Find Excel column name from a given column number](http://www.geeksforgeeks.org/find-excel-column-name-given-number/),[Find next greater number with same set of digits](http://www.geeksforgeeks.org/find-next-greater-number-set-digits/),[Count Possible Decodings of a given Digit Sequence](http://www.geeksforgeeks.org/count-possible-decodings-given-digit-sequence/), [Calculate the angle between hour hand and minute hand](http://www.geeksforgeeks.org/calculate-angle-hour-hand-minute-hand/),  [Count number of binary strings without consecutive 1?s](http://www.geeksforgeeks.org/count-number-binary-strings-without-consecutive-1s/), [Find the smallest number whose digits multiply to a given number n](http://www.geeksforgeeks.org/find-smallest-number-whose-digits-multiply-given-number-n/), [Draw a circle without floating point arithmetic](http://geeksquiz.com/draw-circle-without-floating-point-arithmetic/),

**Bit Algorithms:**[Find the element that appears once](http://www.geeksforgeeks.org/find-the-element-that-appears-once/), [Detect opposite signs](http://www.geeksforgeeks.org/detect-if-two-integers-have-opposite-signs/), [Set bits in all numbers from 1 to n](http://www.geeksforgeeks.org/count-total-set-bits-in-all-numbers-from-1-to-n/), [Swap bits](http://www.geeksforgeeks.org/swap-bits-in-a-given-number/), [Add two numbers](http://www.geeksforgeeks.org/add-two-numbers-without-using-arithmetic-operators/), [Smallest of three](http://www.geeksforgeeks.org/smallest-of-three-integers-without-comparison-operators/), [A Boolean Array Puzzle](http://www.geeksforgeeks.org/a-boolean-array-puzzle/),[Set bits in an (big) array](http://www.geeksforgeeks.org/program-to-count-number-of-set-bits-in-an-big-array/),[Next higher number with same number of set bits](http://www.geeksforgeeks.org/next-higher-number-with-same-number-of-set-bits/), [Optimization Technique (Modulus)](http://www.geeksforgeeks.org/optimization-techniques-set-1-modulus/), [Add 1 to a number](http://www.geeksforgeeks.org/add-1-to-a-given-number/), [Multiply with 3.5](http://www.geeksforgeeks.org/multiply-an-integer-with-3-5/), [Turn off the rightmost set bit](http://www.geeksforgeeks.org/turn-off-the-rightmost-set-bit/), [Check for Power of 4](http://www.geeksforgeeks.org/find-whether-a-given-number-is-a-power-of-4-or-not/), [Absolute value (abs) without branching](http://www.geeksforgeeks.org/compute-the-integer-absolute-value-abs-without-branching/), [Modulus division by a power-of-2-number](http://www.geeksforgeeks.org/compute-modulus-division-by-a-power-of-2-number/),[Minimum or Maximum of two integers](http://www.geeksforgeeks.org/compute-the-minimum-or-maximum-max-of-two-integers-without-branching/), [Rotate bits](http://www.geeksforgeeks.org/rotate-bits-of-an-integer/), [Find the two non-repeating elements in an array](http://www.geeksforgeeks.org/find-two-non-repeating-elements-in-an-array-of-repeating-elements/), [Number Occurring Odd Number of Times](http://www.geeksforgeeks.org/find-the-number-occurring-odd-number-of-times/), [Check for Integer Overflow](http://www.geeksforgeeks.org/check-for-integer-overflow/), [Little and Big Endian](http://www.geeksforgeeks.org/little-and-big-endian-mystery/), [Reverse Bits of a Number](http://www.geeksforgeeks.org/write-an-efficient-c-program-to-reverse-bits-of-a-number/), [Count set bits in an integer](http://www.geeksforgeeks.org/count-set-bits-in-an-integer/), [Number of bits to be flipped to convert A to B](http://www.geeksforgeeks.org/count-number-of-bits-to-be-flipped-to-convert-a-to-b/), [Next Power of 2](http://www.geeksforgeeks.org/next-power-of-2/),[Check if a Number is Multiple of 3](http://www.geeksforgeeks.org/write-an-efficient-method-to-check-if-a-number-is-multiple-of-3/), [Find parity](http://www.geeksforgeeks.org/write-a-c-program-to-find-the-parity-of-an-unsigned-integer/), [Multiply with 7](http://www.geeksforgeeks.org/efficient-way-to-multiply-with-7/), [Find whether a no is power of two](http://www.geeksforgeeks.org/write-one-line-c-function-to-find-whether-a-no-is-power-of-two/), [Position of rightmost set bit](http://www.geeksforgeeks.org/position-of-rightmost-set-bit/), [Binary representation of a given number](http://www.geeksforgeeks.org/binary-representation-of-a-given-number/), [Swap all odd and even bits](http://www.geeksforgeeks.org/swap-all-odd-and-even-bits/), [Find position of the only set bit](http://www.geeksforgeeks.org/find-position-of-the-only-set-bit/), [Karatsuba algorithm for fast multiplication](http://www.geeksforgeeks.org/divide-and-conquer-set-2-karatsuba-algorithm-for-fast-multiplication/" \t "_blank), [How to swap two numbers without using a temporary variable?](http://www.geeksforgeeks.org/swap-two-numbers-without-using-temporary-variable/), [Check if a number is multiple of 9 using bitwise operators](http://www.geeksforgeeks.org/divisibility-9-using-bitwise-operators/)**,**[Swap two nibbles in a byte](http://www.geeksforgeeks.org/swap-two-nibbles-byte/), [How to turn off a particular bit in a number?](http://geeksquiz.com/algorithms/bit-algorithms/), [Check if binary representation of a number is palindrome](http://www.geeksforgeeks.org/check-binary-representation-number-palindrome/)

[Quiz on Bit Algorithms](http://geeksquiz.com/algorithms/bit-algorithms/).

**Graph Algorithms:**  
***Introduction, DFS and BFS:*** [Graph and its representations](http://www.geeksforgeeks.org/graph-and-its-representations/), [Breadth First Traversal for a Graph](http://www.geeksforgeeks.org/breadth-first-traversal-for-a-graph/),[Depth First Traversal for a Graph](http://www.geeksforgeeks.org/depth-first-traversal-for-a-graph/), [Applications of Depth First Search](http://www.geeksforgeeks.org/applications-of-depth-first-search/), [Detect Cycle in a Directed Graph](http://www.geeksforgeeks.org/detect-cycle-in-a-graph/), [Detect Cycle in a an Undirected Graph](http://www.geeksforgeeks.org/union-find/), [Detect cycle in an undirected graph](http://www.geeksforgeeks.org/detect-cycle-undirected-graph/), [Longest Path in a Directed Acyclic Graph](http://www.geeksforgeeks.org/find-longest-path-directed-acyclic-graph/), [Topological Sorting](http://www.geeksforgeeks.org/topological-sorting/), [Check whether a given graph is Bipartite or not](http://www.geeksforgeeks.org/bipartite-graph/), [Snake and Ladder Problem](http://www.geeksforgeeks.org/snake-ladder-problem-2/)

***Minimum Spanning Tree:***[Prim’s Minimum Spanning Tree (MST))](http://www.geeksforgeeks.org/greedy-algorithms-set-5-prims-minimum-spanning-tree-mst-2/), [Applications of Minimum Spanning Tree Problem](http://www.geeksforgeeks.org/applications-of-minimum-spanning-tree/), [Prim’s MST for Adjacency List Representation](http://www.geeksforgeeks.org/greedy-algorithms-set-5-prims-mst-for-adjacency-list-representation/), [Kruskal’s Minimum Spanning Tree Algorithm](http://www.geeksforgeeks.org/greedy-algorithms-set-2-kruskals-minimum-spanning-tree-mst/" \o "Permanent link to Greedy Algorithms | Set 2 (Kruskal’s Minimum Spanning Tree Algorithm))

***Shortest Paths:*** [Dijkstra’s shortest path algorithm](http://www.geeksforgeeks.org/greedy-algorithms-set-6-dijkstras-shortest-path-algorithm/" \o "Permanent link to Greedy Algorithms | Set 7 (Dijkstra’s shortest path algorithm)), [Dijkstra’s Algorithm for Adjacency List Representation](http://www.geeksforgeeks.org/greedy-algorithms-set-7-dijkstras-algorithm-for-adjacency-list-representation/" \o "Permanent link to Greedy Algorithms | Set 8 (Dijkstra’s Algorithm for Adjacency List Representation)), [Bellman–Ford Algorithm](http://www.geeksforgeeks.org/dynamic-programming-set-23-bellman-ford-algorithm/), [Floyd Warshall Algorithm](http://www.geeksforgeeks.org/dynamic-programming-set-16-floyd-warshall-algorithm/), [Johnson’s algorithm for All-pairs shortest paths](http://www.geeksforgeeks.org/johnsons-algorithm/), [Shortest Path in Directed Acyclic Graph](http://www.geeksforgeeks.org/shortest-path-for-directed-acyclic-graphs/),

***Connectivity:***[Find if there is a path between two vertices in a directed graph](http://www.geeksforgeeks.org/find-if-there-is-a-path-between-two-vertices-in-a-given-graph/), [Connectivity in a directed graph](http://www.geeksforgeeks.org/connectivity-in-a-directed-graph/), [Articulation Points (or Cut Vertices) in a Graph](http://www.geeksforgeeks.org/articulation-points-or-cut-vertices-in-a-graph/), [Biconnected graph](http://www.geeksforgeeks.org/biconnectivity-in-a-graph/" \o "Permanent link to Biconnected graph), [Bridges in a graph](http://www.geeksforgeeks.org/bridge-in-a-graph/), [Eulerian path and circuit](http://www.geeksforgeeks.org/eulerian-path-and-circuit/" \o "Permanent link to Eulerian path and circuit), [Fleury’s Algorithm for printing Eulerian Path or Circuit](http://www.geeksforgeeks.org/fleurys-algorithm-for-printing-eulerian-path/" \o "Permanent link to Fleury’s Algorithm for printing Eulerian Path or Circuit), [Strongly Connected Components](http://www.geeksforgeeks.org/strongly-connected-components/), [Transitive closure of a graph](http://www.geeksforgeeks.org/transitive-closure-of-a-graph/), [Find the number of islands](http://www.geeksforgeeks.org/find-number-of-islands/), [Count all possible walks from a source to a destination with exactly k edges](http://www.geeksforgeeks.org/count-possible-paths-source-destination-exactly-k-edges/),  [Euler Circuit in a Directed Graph](http://www.geeksforgeeks.org/euler-circuit-directed-graph/)

***Hard Problems:***[Graph Coloring (Introduction and Applications)](http://www.geeksforgeeks.org/graph-coloring-applications/),[Greedy Algorithm for Graph Coloring](http://www.geeksforgeeks.org/graph-coloring-set-2-greedy-algorithm/), [Travelling Salesman Problem (Naive and Dynamic Programming)](http://www.geeksforgeeks.org/travelling-salesman-problem-set-1/), [Travelling Salesman Problem (Approximate using MST)](http://www.geeksforgeeks.org/travelling-salesman-problem-set-2-approximate-using-mst/), [Hamiltonian Cycle](http://www.geeksforgeeks.org/backtracking-set-7-hamiltonian-cycle/)

***Maximum Flow:*** [Ford-Fulkerson Algorithm for Maximum Flow Problem](http://www.geeksforgeeks.org/ford-fulkerson-algorithm-for-maximum-flow-problem/),[Find maximum number of edge disjoint paths between two vertices](http://www.geeksforgeeks.org/find-edge-disjoint-paths-two-vertices/), [Find minimum s-t cut in a flow network](http://www.geeksforgeeks.org/minimum-cut-in-a-directed-graph/), [Maximum Bipartite Matching](http://www.geeksforgeeks.org/maximum-bipartite-matching/),  [Channel Assignment Problem](http://www.geeksforgeeks.org/channel-assignment-problem/)