this list updated here on Quora:

* Binary Search :  
  [Tutorial, Problems](https://www.topcoder.com/community/data-science/data-science-tutorials/binary-search/),[Tutorial, Implementation](http://geeksquiz.com/binary-search/),[Problem](http://www.spoj.com/problems/AGGRCOW)
* Quicksort :  
  [Tutorial, Implementation](http://geeksquiz.com/quick-sort/),[Tutorial](https://www.topcoder.com/community/competitive-programming/tutorials/sorting)
* Merge Sort :  
  [Tutorial, Implementation](http://geeksquiz.com/merge-sort/),[Tutorial](https://www.topcoder.com/community/competitive-programming/tutorials/sorting)
* Suffix Array :  
  [Tutorial](http://web.stanford.edu/class/cs97si/suffix-array.pdf),[Tutorial, Implementation](http://discuss.codechef.com/questions/21385/a-tutorial-on-suffix-arrays),[Tutorial, Implementation](http://apps.topcoder.com/forums/;jsessionid=BC99925E58CB2628CA9AA3AFC13F6593?module=Thread&start=0&threadID=627379),[Problem](http://www.spoj.com/problems/SUBST1/),[Problem](http://www.codechef.com/problems/MOU1H)
* Knuth-Morris-Pratt Algorithm (KMP) :  
  [Tutorial](https://www.topcoder.com/community/data-science/data-science-tutorials/introduction-to-string-searching-algorithms/),[Tutorial, Implementation](http://www.geeksforgeeks.org/searching-for-patterns-set-2-kmp-algorithm/),[Tutorial](http://keithschwarz.com/interesting/code/?dir=knuth-morris-pratt),[Problem](http://www.codechef.com/problems/TASHIFT)
* Rabin-Karp Algorithm :  
  [Tutorial, Implementation](http://www.geeksforgeeks.org/searching-for-patterns-set-3-rabin-karp-algorithm/),[Tutorial](https://www.topcoder.com/community/competitive-programming/tutorials/introduction-to-string-searching-algorithms/),[Problem](http://www.codechef.com/problems/SSTORY),[Problem](http://codeforces.com/problemset/problem/271/D)
* Tries :  
  [Tutorial, Problems](https://www.topcoder.com/community/competitive-programming/tutorials/using-tries/),[Tutorial : I,](http://www.geeksforgeeks.org/trie-insert-and-search/)[II](http://www.geeksforgeeks.org/trie-delete/),[Tutorial](https://threads-iiith.quora.com/Tutorial-on-Trie-and-example-problems),[Problem](http://www.spoj.com/problems/SUBXOR/),[Problem](https://icpcarchive.ecs.baylor.edu/index.php?Itemid=8&category=345&option=com_onlinejudge&page=show_problem&problem=2683),[Problem](http://www.codechef.com/problems/EST)
* Depth First Traversal of a Graph :  
  [Tutorial, Implementation](http://www.geeksforgeeks.org/depth-first-traversal-for-a-graph/),[Tutorial, Problems](https://www.topcoder.com/community/competitive-programming/tutorials/introduction-to-graphs-and-their-data-structures-section-2/),[Problem](http://www.spoj.com/problems/PARADOX/),[Problem](http://www.spoj.com/problems/BUGLIFE/),[Problem](http://www.spoj.com/problems/PT07Z/)
* Breadth First Traversal of a Graph :  
  [Tutorial, Implementation](http://www.geeksforgeeks.org/breadth-first-traversal-for-a-graph/),[Tutorial, Problems](https://www.topcoder.com/community/competitive-programming/tutorials/introduction-to-graphs-and-their-data-structures-section-2/),[Problem](http://www.codechef.com/problems/DIGJUMP),[Problem](http://www.spoj.com/problems/ONEZERO/),[Problem](http://www.spoj.com/problems/NAKANJ/),[Flood Fill](http://community.topcoder.com/tc?d1=tutorials&d2=findSolution&module=Static#floodfill)
* Dijkstra’s Algorithm :  
  [Tutorial, Problems](https://www.topcoder.com/community/competitive-programming/tutorials/introduction-to-graphs-and-their-data-structures-section-3/),[Problem](http://www.codechef.com/problems/REN2013G),[Tutorial(greedy)](http://e-maxx.ru/algo/dijkstra),[Tutorial (with heap)](http://e-maxx.ru/algo/dijkstra_sparse),[Implementation](http://zobayer.blogspot.in/2009/12/dijkstras-algorithm-in-c.html),[Problem](http://www.spoj.com/problems/EZDIJKST/),[Problem](http://www.spoj.com/problems/SHPATH/)
* Binary Indexed Tree :  
  [Tutorial, Problems](http://community.topcoder.com/tc?d1=tutorials&d2=binaryIndexedTrees&module=Static),[Tutorial](http://codeforces.com/blog/entry/619),[Original Paper](http://citeseerx.ist.psu.edu/viewdoc/download;jsessionid=AB3AEBC0736E52FA815A3D4C633DE52F?doi=10.1.1.14.8917&rep=rep1&type=pdf),[Tutorial](http://sanugupta.wordpress.com/2014/08/29/binary-indexed-tree-fenwick-tree/),[Tutorial](http://cs.stackexchange.com/a/10541),[Problem](http://www.spoj.com/problems/HORRIBLE/),[Problem](http://www.spoj.com/problems/YODANESS/),  
  [Problem](http://www.spoj.com/problems/INVCNT/),[Problem](http://www.spoj.com/problems/NICEDAY/),[Problem](http://www.spoj.com/problems/CTRICK/),[Problem](http://www.spoj.com/problems/DQUERY/),[Problem](http://www.spoj.com/problems/MCHAOS/)
* Segment Tree (with lazy propagation) :  
  [Tutorial, Implementation](http://se7so.blogspot.in/2012/12/segment-trees-and-lazy-propagation.html),[Tutorial](http://discuss.codechef.com/questions/38770/lazy-propagation),[Tutorial, Problems, Implementation](http://letuskode.blogspot.in/2013/01/segtrees.html),[Tutorial, Implementation and Various Uses](http://e-maxx.ru/algo/segment_tree), Persistent Segment Tree: \*[62](http://blog.anudeep2011.com/persistent-segment-trees-explained-with-spoj-problems/),[II](https://discuss.codechef.com/questions/101647/persistence-made-simple-tutorial), problems same as BIT,[Problem](http://www.spoj.com/problems/HORRIBLE/),[Problem](http://www.codechef.com/problems/IDOLS)/HLD is used as well/
* Z algorithm :  
  [Tutorial, Problem](http://codeforces.com/blog/entry/3107),[Tutorial](https://www.cs.umd.edu/class/fall2011/cmsc858s/Lec02-zalg.pdf),[Tutorial](https://ivanyu.me/blog/2013/10/15/z-algorithm/), problems same as KMP.
* Floyd Warshall Algorithm :  
  [Tutorial, Implementation](http://www.geeksforgeeks.org/dynamic-programming-set-16-floyd-warshall-algorithm/),[Problem](http://www.spoj.com/problems/AMR11F/),[Problem](http://community.topcoder.com/stat?c=problem_statement&pm=2356)
* Sparse Table (LCP, RMQ) :  
  [Tutorial, Problems](https://www.topcoder.com/community/data-science/data-science-tutorials/range-minimum-query-and-lowest-common-ancestor/),[Tutorial, Implementation(C++)](http://mayanknatani.wordpress.com/2013/07/15/range-minimum-query/),[Java implementation](https://sites.google.com/site/indy256/algo/sparse_table_rmq)
* Heap / Priority Queue / Heapsort :  
  [Implementation, Explanation](http://www.sourcetricks.com/2011/06/c-heaps.html#.U9z8J_mSzfc),[Tutorial](http://pages.cs.wisc.edu/~vernon/cs367/notes/11.PRIORITY-Q.html),[Implementation](http://www.cprogramming.com/tutorial/computersciencetheory/heapcode.html),[Problem](http://www.codechef.com/problems/REVERSE), Chapter from CLRS
* [Modular Multiplicative Inverse](http://comeoncodeon.wordpress.com/2011/10/09/modular-multiplicative-inverse/)
* Binomial coefficients (nCr % M):[Tutorial](http://discuss.codechef.com/questions/3869/best-known-algos-for-calculating-ncr-m),[Tutorial](http://fishi.devtail.io/weblog/2015/06/25/computing-large-binomial-coefficients-modulo-prime-non-prime/),[Paper](https://www.dropbox.com/s/h7665pcqto17pl4/BinCoeff.pdf) (Link Not Working),[Problem](https://www.codechef.com/problems/SANDWICH)
* Suffix Automaton :  
  [Detailed Paper](http://www.cs.nyu.edu/~mohri/pub/nfac.pdf),[Tutorial, Implementation (I)](http://www.geeksforgeeks.org/searching-for-patterns-set-5-finite-automata/),[Tutorial, Implementation (II)](http://www.geeksforgeeks.org/pattern-searching-set-5-efficient-constructtion-of-finite-automata/),[Problem](http://www.codechef.com/problems/SUBQUERY),[Problem](http://www.codechef.com/problems/TSUBSTR),[Problem](http://www.codechef.com/problems/SSTORY),[Problem](http://www.codechef.com/problems/MOU1H),[Tutorial, Implementation](http://e-maxx.ru/algo/suffix_automata)
* Lowest Common Ancestor :  
  [Tutorial, Problems](http://www.topcoder.com/tc?d1=tutorials&d2=lowestCommonAncestor&module=Static),[Paper](http://www14.informatik.tu-muenchen.de/konferenzen/Jass08/courses/1/moufatich/El_Moufatich_Paper.pdf),[Paper](http://ab.inf.uni-tuebingen.de/people/fischer/lsa.pdf),[Problem](http://www.codechef.com/LTIME14/problems/TALCA),[Problem](http://www.spoj.com/problems/LCA/),[Problem](http://www.codechef.com/problems/TRIPS)
* Counting Inversions :  
  [Divide and Conquer](http://www.geeksforgeeks.org/counting-inversions/),[Segment Tree](https://www.quora.com/How-to-count-inversions-using-Segment-Tree-of-a-given-array),[Fenwick Tree](http://pavelsimo.blogspot.in/2012/09/counting-inversions-in-array-using-BIT.html),[Problem](http://www.codechef.com/problems/DYNAINV)
* [Euclid’s Extended Algorithm](http://discuss.codechef.com/questions/20842/a-tutorial-on-the-extended-euclids-algorithm)
* Suffix Tree :  
  [Tutorial](http://stackoverflow.com/questions/9452701/ukkonens-suffix-tree-algorithm-in-plain-english),[Tutorial](http://marknelson.us/1996/08/01/suffix-trees/),[Intro](http://www.geeksforgeeks.org/pattern-searching-set-8-suffix-tree-introduction/), Construction : \*[106](http://www.geeksforgeeks.org/ukkonens-suffix-tree-construction-part-1/),[II](http://www.geeksforgeeks.org/ukkonens-suffix-tree-construction-part-2/),[Implementation](http://marknelson.us/attachments/1996/suffix-trees/stree2006.cpp),[Implementation](http://www.sanfoundry.com/cpp-program-implement-suffix-tree/),[Problem](http://www.spoj.com/problems/LCS/),[Problem](http://www.codechef.com/OCT11/problems/REPSTR),[Problem](http://www.spoj.com/problems/BEADS/),[Problem](http://www.codechef.com/problems/TASTR)
* Dynamic Programming :  
  Chapter from CLRS(essential),[Tutorial, Problems](https://www.topcoder.com/community/competitive-programming/tutorials/dynamic-programming-from-novice-to-advanced/),[Problem](http://www.codechef.com/problems/LEPAINT),[Problem](http://www.codechef.com/problems/COINS),[Problem](http://www.codechef.com/problems/MARCHA1),[Problem](http://discuss.codechef.com/questions/47239/frogv-editorial),[Tutorial](https://www.quora.com/Are-there-any-good-resources-or-tutorials-for-dynamic-programming-DP-besides-the-TopCoder-tutorial),[Problem](http://www.codechef.com/problems/TSHIRTS),[Problem](http://community.topcoder.com/stat?c=problem_statement&pm=11566),[Problem](http://www.spoj.com/problems/SOCOLA/),[Longest Increasing Subsequence](http://www.geeksforgeeks.org/longest-monotonically-increasing-subsequence-size-n-log-n/),[Bitmask DP](http://codeforces.com/blog/entry/337),[Bitmask DP](http://www.ugrad.cs.ubc.ca/~cs490/sec202/notes/dp/DP%202.pdf),[Optimization](http://codeforces.com/blog/entry/8219),[Problem](http://www.spoj.com/problems/TRSTAGE/),[Problem](http://www.spoj.com/problems/LAZYCOWS/),[Problem](http://www.spoj.com/problems/HIST2/),[Problem](http://www.spoj.com/problems/MKPAIRS/),[Problem](http://www.spoj.com/problems/NKLEAVES/),[Problem](http://www.spoj.com/problems/DRAGON2/),[Problem](http://codeforces.com/contest/461/problem/B), DP on Trees : \*[134](http://www.iarcs.org.in/inoi/online-study-material/topics/dp-trees.php),[II](http://www.cs.berkeley.edu/~vazirani/s99cs170/notes/dynamic2.pdf)
* Basic Data Structures :  
  [Tutorial](https://www.topcoder.com/community/competitive-programming/tutorials/data-structures/),[Stack Implementation](https://www.cs.bu.edu/teaching/c/stack/array/),[Queue Implementation, Tutorial](http://geeksquiz.com/queue-set-1introduction-and-array-implementation/),[Linked List Implementation](http://codingfreak.blogspot.com/2009/08/implementation-of-singly-linked-list-in.html)
* [Logarithmic Exponentiation](http://discuss.codechef.com/questions/20451/a-tutorial-on-fast-modulo-multiplication-exponential-squaring)
* Graphs :  
  [Definition, Representation](http://discuss.codechef.com/questions/17801/introduction-to-graphs-definitions-traversal-depth-first-search),[Definition, Representation](https://www.topcoder.com/community/competitive-programming/tutorials/introduction-to-graphs-and-their-data-structures-section-1/),[Problem](http://www.codechef.com/problems/DRGHTS),[Problem](http://www.codechef.com/problems/DIREL)
* Minimum Spanning Tree :  
  [Tutorial](https://www.ics.uci.edu/~eppstein/161/960206.html),[Tutorial, Kruskal’s Implementation](http://www.geeksforgeeks.org/greedy-algorithms-set-2-kruskals-minimum-spanning-tree-mst/),[Prim’s Implementation](http://www.geeksforgeeks.org/greedy-algorithms-set-5-prims-minimum-spanning-tree-mst-2/),[Problem](http://www.spoj.com/problems/MST/),[Problem](http://www.spoj.com/problems/CSTREET/),[Problem](http://www.spoj.com/problems/BLINNET/),[Problem](http://community.topcoder.com/stat?c=problem_statement&pm=7921&rd=10765),[Problem](http://community.topcoder.com/stat?c=problem_statement&pm=7643&rd=12058)
* [Efficient Prime Factorization](http://www.geeksforgeeks.org/print-all-prime-factors-of-a-given-number/)
* Combinatorics :  
  [Tutorial, Problems](https://www.topcoder.com/community/competitive-programming/tutorials/basics-of-combinatorics/),[Problem](http://www.codechef.com/problems/BINTOUR),[Tutorial](http://apps.topcoder.com/forums/?mc=13&module=Thread&start=0&threadID=334598#335550)
* Union Find/Disjoint Set :  
  [Tutorial](http://www.cs.cornell.edu/~wdtseng/icpc/notes/graph_part4.pdf),[Tutorial, Problems](http://community.topcoder.com/tc?d1=tutorials&d2=disjointDataStructure&module=Static),[Problem](http://www.codechef.com/problems/DISHOWN),[Problem](http://www.spoj.com/problems/BLINNET/),[Problem](http://www.spoj.com/problems/CHAIN/)
* Knapsack problem :  
  [Solution, Implementation](http://www.geeksforgeeks.org/dynamic-programming-set-10-0-1-knapsack-problem/)
* Aho-Corasick String Matching Algorithm :  
  [Tutorial](http://www.cs.sun.ac.za/~lvzijl/courses/rw778/autappl/crous-hw2.pdf),[Implementation](https://gist.github.com/andmej/1233426),[Problem](http://www.codechef.com/problems/FAVNUM),[Problem](http://community.topcoder.com/stat?c=problem_statement&pm=11514&rd=14544),[Problem](http://community.topcoder.com/stat?c=problem_statement&pm=6017),[Problem](http://www.spoj.com/problems/WPUZZLES/)
* Strongly Connected Components :  
  [Tutorial, Implementation](http://www.geeksforgeeks.org/strongly-connected-components/),[Tutorial](http://www.cs.berkeley.edu/~vazirani/s99cs170/notes/lec12.pdf),[Problem](http://www.spoj.com/problems/BOTTOM/),[Problem](http://www.spoj.com/problems/BREAK/),[Problem](http://community.topcoder.com/stat?c=problem_statement&pm=8488&rd=11125)
* Bellman Ford algorithm :  
  [Tutorial, Implementation](http://www.geeksforgeeks.org/dynamic-programming-set-23-bellman-ford-algorithm/),[Tutorial, Implementation](http://compprog.wordpress.com/2007/11/29/one-source-shortest-path-the-bellman-ford-algorithm/),[Problem](http://community.topcoder.com/stat?c=problem_statement&pm=10580),[Problem](http://codeforces.com/problemset/problem/346/D)
* Heavy-light Decomposition :  
  [Tutorial, Problems](http://e-maxx.ru/algo/heavy_light),[Tutorial, Implementation](http://blog.anudeep2011.com/heavy-light-decomposition/),[Tutorial](http://wcipeg.com/wiki/Heavy-light_decomposition),[Implementation](http://apps.topcoder.com/forums/?mc=8&module=Thread&start=0&threadID=796128),[Implementation](http://pastie.org/private/ozpqitws20ylrj8a57tog),[Problem](http://www.spoj.com/problems/QTREE6/),[Problem](http://www.codechef.com/problems/PUSHFLOW),[Problem](http://www.codechef.com/problems/GERALD2)
* Convex Hull :  
  [Tutorial, Jarvis Algorithm Implementation](http://www.geeksforgeeks.org/convex-hull-set-1-jarviss-algorithm-or-wrapping/),[Tutorial with Graham scan](http://www.geeksforgeeks.org/convex-hull-set-2-graham-scan/),[Tutorial](https://www.topcoder.com/community/data-science/data-science-tutorials/geometry-concepts-line-intersection-and-its-applications/),[Implementation](http://stanford.edu/~liszt90/acm/notebook.html#file8),[Problem](https://www.topcoder.com/stat?c=problem_statement&pm=3996&rd=7224),[Problem](http://codeforces.com/problemset/problem/166/B),[Problem](http://community.topcoder.com/stat?c=problem_statement&pm=1960&rd=4670),[Problem](http://acm.timus.ru/problem.aspx?num=1185&space=1),[Problem](http://www.spoj.com/problems/BSHEEP/)
* Line Intersection :  
  [Tutorial, Implementation](http://www.geeksforgeeks.org/check-if-two-given-line-segments-intersect/),[Tutorial, Problems](https://www.topcoder.com/community/competitive-programming/tutorials/geometry-concepts-line-intersection-and-its-applications/)
* [Sieve of Erastothenes](http://www.geeksforgeeks.org/sieve-of-eratosthenes/)
* Interval Tree :  
  [Tutorial, Implementation](http://www.geeksforgeeks.org/interval-tree/),[Problem](http://www.codechef.com/problems/FLIPCOIN/),[Problem](http://www.spoj.com/problems/THRBL/),[Problem](http://www.spoj.com/problems/LITE/),[Problem](http://www.spoj.com/problems/FREQUENT/),[Problem](http://www.spoj.com/problems/GSS1/),[Problem](http://www.spoj.com/problems/GSS3/),[Tutorial](http://www.dgp.toronto.edu/people/JamesStewart/378notes/22intervals/)
* [Counting Sort](http://www.geeksforgeeks.org/counting-sort/)
* [Probabilities](https://www.topcoder.com/community/competitive-programming/tutorials/understanding-probabilities/)
* Matrix Exponentiation :  
  [Tutorial](http://discuss.codechef.com/questions/2335/building-up-the-recurrence-matrix-to-compute-recurrences-in-ologn-time),[Tutorial](http://zobayer.blogspot.in/2010/11/matrix-exponentiation.html)
* Network flow :  
  [(Max Flow)Tutorial : I](https://www.topcoder.com/community/competitive-programming/tutorials/maximum-flow-section-1/)[,](http://help.topcoder.com/data-science/competing-in-algorithm-challenges/algorithm-tutorials/maximum-flow-section-1/)[II](http://community.topcoder.com/tc?d1=tutorials&d2=maxFlow2&module=Static),[Max Flow(Ford-Fulkerson) Tutorial, Implementation](http://www.geeksforgeeks.org/ford-fulkerson-algorithm-for-maximum-flow-problem/),[(Min Cut) Tutorial, Implementation](http://www.geeksforgeeks.org/minimum-cut-in-a-directed-graph/),[(Min Cost Flow)Tutorial : I,](http://community.topcoder.com/tc?d1=tutorials&d2=minimumCostFlow1&module=Static)[II](https://www.topcoder.com/community/competitive-programming/tutorials/minimum-cost-flow-part-two-algorithms/)[,](http://help.topcoder.com/data-science/competing-in-algorithm-challenges/algorithm-tutorials/minimum-cost-flow-part-2-algorithms/)[III](https://www.topcoder.com/community/competitive-programming/tutorials/minimum-cost-flow-part-three-applications/),[Dinic’s Algorithm with Implementation](http://e-maxx.ru/algo/dinic),[Max flow by Edmonds Karp with Implementation](http://e-maxx.ru/algo/edmonds_karp),[Problem](http://www.codechef.com/problems/TWOCOMP),[Problem](http://www.codechef.com/problems/LONGART),[Problem](http://www.codechef.com/problems/ANUBTT),[Problem](http://www.codechef.com/problems/ORDERAAM),[Problem](http://www.codechef.com/problems/PARADE),[Problem](http://www.codechef.com/problems/CAKE2AM),[Problem](http://www.spoj.com/problems/EN/),[Problem](http://www.spoj.com/problems/POTHOLE/),[Problem](http://www.spoj.com/problems/SCITIES/),[Problem](http://www.spoj.com/problems/GREED/),[Problem](http://www.spoj.com/problems/TOURS/),[Problem](http://community.topcoder.com/stat?c=problem_statement&pm=1931&rd=4709),[Problem](http://community.topcoder.com/stat?c=problem_statement&pm=2852&rd=5075),[Problem](http://community.topcoder.com/stat?c=problem_statement&pm=3530&rd=6535)
* K-d tree :  
  [Tutorial](http://web.stanford.edu/class/cs106l/handouts/assignment-3-kdtree.pdf),[Tutorial](http://www.autonlab.org/autonweb/14665/version/2/part/5/data/moore-tutorial.pdf?branch=main&language=en),[Implementation](http://rosettacode.org/wiki/K-d_tree),[Problem](http://www.spoj.com/problems/GANNHAT/)
* [Deque](http://www.sourcetricks.com/2011/06/c-deque.html#.U--v__mSzfc)
* Binary Search Tree :  
  [Tutorial, Implementation](http://www.sourcetricks.com/2011/06/binary-search-trees-in-c.html#.U--wAvmSzfc),[Searching and Insertion](http://geeksquiz.com/binary-search-tree-set-1-search-and-insertion/),[Deletion](http://geeksquiz.com/binary-search-tree-set-2-delete/)
* Quick Select :  
  [Implementation](http://www.sourcetricks.com/2011/06/quick-select.html#.U_CQ0_mSzfc),[Implementation](http://rosettacode.org/wiki/Quickselect_algorithm#C.2B.2B)
* Treap/Cartesian Tree :  
  [Tutorial(detailed)](http://habrahabr.ru/post/101818/),[Tutorial, Implementation](http://e-maxx.ru/algo/treap),[Uses and Problems](http://codeforces.com/blog/entry/3767),[Problem](http://www.codechef.com/problems/CARDSHUF/),[Problem](http://www.codechef.com/problems/CHEFC)
* Game Theory :  
  [Detailed Paper](http://www.math.ucla.edu/~tom/Game_Theory/comb.pdf),[Tutorial, Problems](http://community.topcoder.com/tc?d1=tutorials&d2=algorithmGames&module=Static),[Grundy Numbers](http://letuskode.blogspot.ch/2014/08/grundy-numbers.html),[Tutorial with example problems - I,](http://www.thelearningpoint.net/home/mathematics/an-introduction-to-game-theory)[II,](http://www.thelearningpoint.net/home/mathematics/a-totorial-on-extensive-games-with-problems-and-solutions)[III,](http://www.thelearningpoint.net/home/mathematics/bayesian-games---games-with-incomplete-information)[IV](http://www.thelearningpoint.net/home/mathematics/repeated-games---tutorial-and-solved-problems),[Tutorial, Problems](http://www.codechef.com/wiki/tutorial-game-theory),[Problem](http://www.spoj.com/problems/NGM/),[Problem](http://www.spoj.com/problems/MCOINS/),[Problem](http://www.spoj.com/problems/QCJ3/),[Problem](http://www.spoj.com/problems/RESN04/),[Problem](http://www.spoj.com/problems/MMMGAME/),[Problem](http://www.spoj.com/problems/PEBBMOV/),[Problem](http://www.codechef.com/problems/CHEFBRO),[Problem](http://www.spoj.com/problems/HUBULLU/),[Problem](http://www.codechef.com/problems/BIGPIZA),[Problem](http://codeforces.com/contest/87/problem/C),[Problem](http://www.spoj.com/problems/CRSCNTRY/),[Nim](http://codeforces.com/blog/entry/3657)
* STL (C++) :  
  [I](https://www.topcoder.com/community/competitive-programming/tutorials/power-up-c-with-the-standard-template-library-part-1/)[,](http://help.topcoder.com/data-science/competing-in-algorithm-challenges/algorithm-tutorials/power-up-c-with-the-standard-template-library-part-i/)[II](https://www.topcoder.com/community/competitive-programming/tutorials/power-up-c-with-the-standard-template-library-part-2/),[Crash Course](http://community.topcoder.com/tc?d1=features&d2=082803&module=Static)
* [Maximum Bipartite Matching](http://www.geeksforgeeks.org/maximum-bipartite-matching/)
* Manacher’s Algorithm :  
  [Implementation](http://leetcode.com/2011/11/longest-palindromic-substring-part-ii.html),[Tutorial](http://tarokuriyama.com/projects/palindrome2.php),[Tutorial, Implementation](http://tristan-interview.blogspot.in/2011/11/longest-palindrome-substring-manachers.html),[Tutorial, Implementation](http://e-maxx.ru/algo/palindromes_count),[Problem](http://acm.timus.ru/problem.aspx?num=1937&space=1),[Problem](http://www.spoj.com/problems/LPS/),[Problem](http://www.spoj.com/problems/MSUBSTR/)
* [Miller-Rabin Primality Test](http://community.topcoder.com/tc?d1=tutorials&d2=primalityTesting&module=Static) :[Code](http://rosettacode.org/wiki/Miller-Rabin_primality_test#C)
* [Stable Marriage Problem](http://www.geeksforgeeks.org/stable-marriage-problem/)
* [Hungarian Algorithm](http://community.topcoder.com/tc?d1=tutorials&d2=hungarianAlgorithm&module=Static),[Tutorial](https://math.uc.edu/~halpern/Linear.progr.folder/Handouts.lp.02/Hungarian.algorithm.pdf)
* [Sweep line Algorithm : I](https://www.topcoder.com/tc?d1=tutorials&d2=lineSweep&module=Static),[II](http://www.geeksforgeeks.org/given-a-set-of-line-segments-find-if-any-two-segments-intersect/)
* LCP :  
  [Tutorial, Implementation](http://codeforces.com/blog/entry/12796#comment-175287),[Tutorial, Implementation](http://e-maxx.ru/algo/suffix_array#7)
* [Gaussian Elimination](http://compprog.wordpress.com/2007/12/11/gaussian-elimination/)
* [Pollard Rho Integer Factorization](http://www.cs.colorado.edu/~srirams/classes/doku.php/pollard_rho_tutorial),[problem](http://www.spoj.com/problems/FACT1/)
* [Topological Sorting](http://www.geeksforgeeks.org/topological-sorting/)
* Detecting Cycles in a Graph : Directed - \*[293](http://www.geeksforgeeks.org/detect-cycle-in-a-graph/),[II](http://www.geeksforgeeks.org/union-find/)Undirected : \*[295](http://www.geeksforgeeks.org/detect-cycle-undirected-graph/)
* Geometry :[Basics](https://www.topcoder.com/community/competitive-programming/tutorials/geometry-concepts-basic-concepts/),[Tutorial](http://web.stanford.edu/class/cs97si/09-computational-geometry.pdf)
* Backtracking :  
  [N queens problem](http://www.geeksforgeeks.org/backtracking-set-3-n-queen-problem/),[Tug of War](http://www.geeksforgeeks.org/tug-of-war/),[Sudoku](http://www.geeksforgeeks.org/backtracking-set-7-suduku/)
* Eulerian and Hamiltonian Paths :  
  [Tutorial](http://www.cs.sfu.ca/~ggbaker/zju/math/euler-ham.html#ham),[Tutorial](http://www.csd.uoc.gr/~hy583/papers/ch14.pdf),[(Eulerian Path and Cycle)Implementation](http://www.geeksforgeeks.org/eulerian-path-and-circuit/),[(Hamiltonian Cycle)Implementation](http://www.geeksforgeeks.org/backtracking-set-7-hamiltonian-cycle/)
* Graph Coloring :  
  [Tutorial, Implementation](http://algorithm.daqwest.com/search?search=Coloring+algorithm)
* Meet in the Middle :  
  [Tutorial](http://www.infoarena.ro/blog/meet-in-the-middle),[Implementation](https://sites.google.com/site/indy256/algo/meet-in-the-middle)
* [Arbitrary Precision Integer(BigInt)](http://pastebin.com/aQ8NJ197),[II](https://github.com/anudeep2011/programming/blob/master/bigint.cpp)
* [Radix Sort](http://www.geeksforgeeks.org/radix-sort/),[Bucket Sort](http://www.geeksforgeeks.org/bucket-sort-2/)
* Johnson’s Algorithm :  
  [Tutorial](http://www.geeksforgeeks.org/johnsons-algorithm/),[Tutorial](http://en.wikipedia.org/wiki/Johnson's_algorithm),[Implementation](https://gist.github.com/ashleyholman/6793360)
* Maximal Matching in a General Graph :  
  [Blossom/Edmond’s Algorithm, Implementation](http://e-maxx.ru/algo/matching_edmonds),[Tutte Matrix](http://e-maxx.ru/algo/tutte_matrix),[Problem](http://www.codechef.com/problems/SEAGRP)
* Recursion :[I,](http://community.topcoder.com/tc?d1=tutorials&d2=recursionPt1&module=Static)[II](http://community.topcoder.com/tc?d1=tutorials&d2=recursionPt2&module=Static),[Towers of Hanoi](http://geeksquiz.com/c-program-for-tower-of-hanoi/) with[explanation](http://en.wikipedia.org/wiki/Tower_of_Hanoi#Recursive_solution)
* [Inclusion and Exclusion Principle : I](http://apps.topcoder.com/forums/?module=Thread&start=0&threadID=685138),[II](http://e-maxx.ru/algo/inclusion_exclusion_principle)
* [Co-ordinate Compression](https://www.quora.com/What-is-coordinate-compression-and-what-is-it-used-for)
* Sqrt-Decomposition :  
  [Tutorial](http://e-maxx.ru/algo/sqrt_decomposition),[Tutorial](http://sysmagazine.com/posts/138946/),[Problem](http://www.spoj.com/problems/RACETIME/),[Problem](http://www.codechef.com/problems/GERALD07)
* Link-Cut Tree :  
  [Tutorial](http://www.cs.cmu.edu/~sleator/papers/dynamic-trees.pdf),[Wiki](http://en.wikipedia.org/wiki/Link/cut_tree),[Tutorial, Implementation](http://www.cs.cmu.edu/~avrim/451f12/lectures/lect1009-linkcut.txt),[Problem](http://www.codechef.com/problems/QTREE6),[Problem](http://www.spoj.com/problems/DYNACON1/),[Problem](http://www.spoj.com/problems/DYNALCA/),[Problem](http://codeforces.com/contest/117/problem/E)
* Euler’s Totient Function :  
  [Explanation, Implementation, Problems](http://e-maxx.ru/algo/euler_function),[Explanation, Problems](http://community.topcoder.com/tc?d1=tutorials&d2=primeNumbers&module=Static)
* Burnside Lemma :  
  [Tutorial](http://e-maxx.ru/algo/burnside_polya),[Tutorial](http://petr-mitrichev.blogspot.in/2008/11/burnsides-lemma.html),[Problem](http://community.topcoder.com/stat?c=problem_statement&pm=9975)
* Edit/Levenshtein Distance :  
  [Tutorial](https://web.stanford.edu/class/cs124/lec/med.pdf),[Introduction](http://en.wikipedia.org/wiki/Wagner%E2%80%93Fischer_algorithm),[Tutorial](http://www.csse.monash.edu.au/~lloyd/tildeAlgDS/Dynamic/Edit/),[Problem](http://www.codechef.com/problems/SEATSR),[Problem](http://www.spoj.com/problems/EDIST/)
* [Branch and Bound](http://www.academic.marist.edu/~jzbv/algorithms/Branch%20and%20Bound.htm)
* [Math for Competitive Programming](http://community.topcoder.com/tc?d1=tutorials&d2=math_for_topcoders&module=Static)
* Mo’s Algorithm :[Tutorial and Problems](http://blog.anudeep2011.com/mos-algorithm/)