**ARRAYS**

* [~~Find 2 elements with given sum~~](http://www.geeksforgeeks.org/write-a-c-program-that-given-a-set-a-of-n-numbers-and-another-number-x-determines-whether-or-not-there-exist-two-elements-in-s-whose-sum-is-exactly-x/)
* [~~Majority Element~~](http://www.geeksforgeeks.org/majority-element/)
* [~~Find the number occuring odd number of times~~](http://www.geeksforgeeks.org/find-the-number-occurring-odd-number-of-times/)
* Memory Checkpoint
* [~~Merge an array of size n into another of size m + n~~](http://www.geeksforgeeks.org/merge-one-array-of-size-n-into-another-one-of-size-mn/)
* [Rotate an array](http://www.geeksforgeeks.org/program-for-array-rotation-continued-reversal-algorithm/)
* [Leaders in an array](http://www.geeksforgeeks.org/leaders-in-an-array/)
* [Majority element in sorted array](http://www.geeksforgeeks.org/check-for-majority-element-in-a-sorted-array/)
* [Segregate 0s and 1s in an array](http://www.geeksforgeeks.org/segregate-0s-and-1s-in-an-array-by-traversing-array-once/)
* [Product array](http://www.geeksforgeeks.org/a-product-array-puzzle/)
* [Find 2 repeating elements](http://www.geeksforgeeks.org/find-the-two-repeating-elements-in-a-given-array/)
* [Find the smallest missing number](http://www.geeksforgeeks.org/find-the-first-missing-number/)
* [Find max j-i such that arr[j] > arr[i]](http://www.geeksforgeeks.org/given-an-array-arr-find-the-maximum-j-i-such-that-arrj-arri/)
* [Find subarray with given sum](http://www.geeksforgeeks.org/find-subarray-with-given-sum/)
* [Find the smallest positive number missing from an unsorted array](http://www.geeksforgeeks.org/find-the-smallest-positive-number-missing-from-an-unsorted-array/)
* [Find 2 numbers with odd occurence](http://www.geeksforgeeks.org/find-the-two-numbers-with-odd-occurences-in-an-unsorted-array/)
* [Largest subarray with equal number of 0s and 1s](http://www.geeksforgeeks.org/largest-subarray-with-equal-number-of-0s-and-1s/)
* [Replace every element with the greatest on right side](http://www.geeksforgeeks.org/replace-every-element-with-the-greatest-on-right-side/)
* [Stock buy sell to maximize profit](http://www.geeksforgeeks.org/stock-buy-sell/)
* [Find common elements in 3 sorted arrays](http://www.geeksforgeeks.org/find-common-elements-three-sorted-arrays/)
* [Nuts and bolts problem](http://www.geeksforgeeks.org/nuts-bolts-problem-lock-key-problem/)
* [Trapping rain water](http://www.geeksforgeeks.org/trapping-rain-water/)
* [Merge 2 sorted arrays in O(1) space](http://www.geeksforgeeks.org/merge-two-sorted-arrays-o1-extra-space/)

**STRINGS**

[Remove duplicates from string](http://www.geeksforgeeks.org/remove-all-duplicates-from-the-input-string/)

[Remove characters from the first string which are present in the second string](http://www.geeksforgeeks.org/remove-characters-from-the-first-string-which-are-present-in-the-second-string/)

[Check if strings are rotations of each other](http://www.geeksforgeeks.org/a-program-to-check-if-strings-are-rotations-of-each-other-or-not/)

[Print all permutations of a given string](http://www.geeksforgeeks.org/write-a-c-program-to-print-all-permutations-of-a-given-string/)

[Reverse words in a given string](http://www.geeksforgeeks.org/reverse-words-in-a-given-string/)

[Find the smallest window in a string containing all the characters of the second string](http://www.geeksforgeeks.org/find-the-smallest-window-in-a-string-containing-all-characters-of-another-string/)

[Check whether two strings are anagrams of each other](http://www.geeksforgeeks.org/check-whether-two-strings-are-anagram-of-each-other/)

[Write your own atoi()](http://www.geeksforgeeks.org/write-your-own-atoi/)

[Rearrange a string so that similar characters become d distance away](http://www.geeksforgeeks.org/rearrange-a-string-so-that-all-same-characters-become-at-least-d-distance-away/)

[Find excel column name from a given column number](http://www.geeksforgeeks.org/find-excel-column-name-given-number/)

**LINKED LIST**

[Get Nth node in a linked list](http://www.geeksforgeeks.org/write-a-function-to-get-nth-node-in-a-linked-list/)

[Delete a node given a pointer to it](http://www.geeksforgeeks.org/given-only-a-pointer-to-a-node-to-be-deleted-in-a-singly-linked-list-how-do-you-delete-it/)

[Print middle](http://www.geeksforgeeks.org/write-a-c-function-to-print-the-middle-of-the-linked-list/)

[Find Nth node from the end](http://www.geeksforgeeks.org/nth-node-from-the-end-of-a-linked-list/)

[Delete linked list](http://www.geeksforgeeks.org/write-a-function-to-delete-a-linked-list/)

[Reverse linked list](http://www.geeksforgeeks.org/write-a-function-to-reverse-the-nodes-of-a-linked-list/)

[Detect loop in a linked list](http://www.geeksforgeeks.org/write-a-c-function-to-detect-loop-in-a-linked-list/) - McKinsey

[Check if a singly linked list is a palindrome](http://www.geeksforgeeks.org/function-to-check-if-a-singly-linked-list-is-palindrome/)

[Clone a linked list with next and random pointer](http://www.geeksforgeeks.org/a-linked-list-with-next-and-arbit-pointer/)

[Memory efficient doubly linked list](http://www.geeksforgeeks.org/memory-efficient-doubly-linked-list/)

[Insert in sorted linked list](http://www.geeksforgeeks.org/given-a-linked-list-which-is-sorted-how-will-you-insert-in-sorted-way/)

[Get intersection point of 2 linked lists](http://www.geeksforgeeks.org/write-a-function-to-get-the-intersection-point-of-two-linked-lists/)

[Print reverse of a linked list](http://www.geeksforgeeks.org/write-a-recursive-function-to-print-reverse-of-a-linked-list/)

[Remove duplicates from sorted linked list](http://www.geeksforgeeks.org/remove-duplicates-from-a-sorted-linked-list/)

[Remove duplicates from unsorted linked list](http://www.geeksforgeeks.org/remove-duplicates-from-an-unsorted-linked-list/)

[Reverse doubly linked list](http://www.geeksforgeeks.org/reverse-a-doubly-linked-list/)

[Merge 2 sorted linked lists](http://www.geeksforgeeks.org/merge-two-sorted-linked-lists/)

[Merge sort for linked lists](http://www.geeksforgeeks.org/merge-sort-for-linked-list/)

[Reverse a linked list in groups of given size](http://www.geeksforgeeks.org/reverse-a-list-in-groups-of-given-size/)

[Linked list vs Array](http://www.geeksforgeeks.org/linked-list-vs-array/)

[Sorted insert for circular linked list](http://www.geeksforgeeks.org/sorted-insert-for-circular-linked-list/)

[Detect and remove loop in a linked list](http://www.geeksforgeeks.org/detect-and-remove-loop-in-a-linked-list/)

[Add 2 numbers represented by linked lists](http://www.geeksforgeeks.org/add-two-numbers-represented-by-linked-lists/)

[Clone a linked list with next and random pointer | Set 2](http://www.geeksforgeeks.org/clone-linked-list-next-arbit-pointer-set-2/)

**TREES**

[Recursive Tree Traversals](http://www.geeksforgeeks.org/618/)

[Calculate size of tree](http://www.geeksforgeeks.org/write-a-c-program-to-calculate-size-of-a-tree/)

[Check if two trees are identical](http://www.geeksforgeeks.org/write-c-code-to-determine-if-two-trees-are-identical/)

[Height of tree](http://www.geeksforgeeks.org/write-a-c-program-to-find-the-maximum-depth-or-height-of-a-tree/)

[Delete a tree](http://www.geeksforgeeks.org/write-a-c-program-to-delete-a-tree/)

[Convert a binary tree to its mirror tree](http://www.geeksforgeeks.org/write-an-efficient-c-function-to-convert-a-tree-into-its-mirror-tree/)

[Given two traversal sequences, construct the binary tree](http://www.geeksforgeeks.org/if-you-are-given-two-traversal-sequences-can-you-construct-the-binary-tree/)

[Print all root to leaf paths in a binary tree](http://www.geeksforgeeks.org/given-a-binary-tree-print-out-all-of-its-root-to-leaf-paths-one-per-line/)

[Lowest common ancestor in BST](http://www.geeksforgeeks.org/lowest-common-ancestor-in-a-binary-search-tree/)

[Level order traversal](http://www.geeksforgeeks.org/level-order-tree-traversal/)

[Count leaf nodes](http://www.geeksforgeeks.org/write-a-c-program-to-get-count-of-leaf-nodes-in-a-binary-tree/)

[Spiral level order traversal](http://www.geeksforgeeks.org/level-order-traversal-in-spiral-form/)

[Diameter of tree](http://www.geeksforgeeks.org/diameter-of-a-binary-tree/)

[Inorder traversal without recursion](http://www.geeksforgeeks.org/inorder-tree-traversal-without-recursion/)

[Root to leaf path sum equal to given number](http://www.geeksforgeeks.org/root-to-leaf-path-sum-equal-to-a-given-number/)

[Construct tree from inorder and preorder traversal](http://www.geeksforgeeks.org/construct-tree-from-given-inorder-and-preorder-traversal/)

[Print nodes at k distance from root](http://www.geeksforgeeks.org/print-nodes-at-k-distance-from-root/)

[Applications of tree](http://www.geeksforgeeks.org/applications-of-tree-data-structure/)

[Check if a binary tree is a subtree of another binary tree](http://www.geeksforgeeks.org/check-if-a-binary-tree-is-subtree-of-another-binary-tree/)

[Find inorder successor for all nodes](http://www.geeksforgeeks.org/populate-inorder-successor-for-all-nodes/)

[Vertical sum in a given binary tree](http://www.geeksforgeeks.org/vertical-sum-in-a-given-binary-tree/)

[Maximum sum root to leaf path](http://www.geeksforgeeks.org/find-the-maximum-sum-path-in-a-binary-tree/)

[Check if a binary tree is complete or not](http://www.geeksforgeeks.org/check-if-a-given-binary-tree-is-complete-tree-or-not/)

[Iterative preorder traversal](http://www.geeksforgeeks.org/iterative-preorder-traversal/)

[Iterative postorder traversal](http://www.geeksforgeeks.org/iterative-postorder-traversal-using-stack/)

[Reverse level order traversal](http://www.geeksforgeeks.org/reverse-level-order-traversal/)

[Binary tree to doubly linked list](http://www.geeksforgeeks.org/in-place-convert-a-given-binary-tree-to-doubly-linked-list/) OR [Binary tree to double linked list](http://www.geeksforgeeks.org/convert-given-binary-tree-doubly-linked-list-set-3/)

[Find height of tree iteratively](http://www.geeksforgeeks.org/iterative-method-to-find-height-of-binary-tree/)

[Left view of binary tree](http://www.geeksforgeeks.org/print-left-view-binary-tree/)

[Lowest common ancestor binary tree](http://www.geeksforgeeks.org/lowest-common-ancestor-binary-tree-set-1/)

[Print all nodes at k distance from given node](http://www.geeksforgeeks.org/print-nodes-distance-k-given-node-binary-tree/)

[Right view of binary tree](http://www.geeksforgeeks.org/print-right-view-binary-tree-2/)

[Check if binary tree is subtree of another binary tree](http://www.geeksforgeeks.org/check-binary-tree-subtree-another-binary-tree-set-2/)

[Print nodes b/w two given levels](http://www.geeksforgeeks.org/given-binary-tree-print-nodes-two-given-level-numbers/)

[Find node with min value in BST](http://www.geeksforgeeks.org/find-the-minimum-element-in-a-binary-search-tree/)

[Check if a binary tree is BST](http://www.geeksforgeeks.org/a-program-to-check-if-a-binary-tree-is-bst-or-not/)

[Find kth smallest element in BST](http://www.geeksforgeeks.org/find-k-th-smallest-element-in-bst-order-statistics-in-bst/)

[Sorted linked list to balanced BST](http://www.geeksforgeeks.org/sorted-linked-list-to-balanced-bst/)

[Kth largest element in BST](http://www.geeksforgeeks.org/kth-largest-element-in-bst-when-modification-to-bst-is-not-allowed/)

[Advantages of BST over hash table](http://www.geeksforgeeks.org/advantages-of-bst-over-hash-table/)

[Kth smallest element in BST using O(1) space](http://www.geeksforgeeks.org/kth-largest-element-in-bst-using-o1-extra-space/)

**STACK**

[Implement queue using stack](http://www.geeksforgeeks.org/queue-using-stacks/)

[Check for balanced parentheses in an expression](http://www.geeksforgeeks.org/check-for-balanced-parentheses-in-an-expression/)

[Reverse a string using recursion](http://www.geeksforgeeks.org/reverse-a-stack-using-recursion/)

[Design and implement special stack](http://www.geeksforgeeks.org/design-and-implement-special-stack-data-structure/)

[Implement stack using queues](http://www.geeksforgeeks.org/implement-stack-using-queue/)

[Expression evaluation](http://www.geeksforgeeks.org/expression-evaluation/)

**GRAPH**

[Applications of MST](http://www.geeksforgeeks.org/applications-of-minimum-spanning-tree/)

[Applications of DFS](http://www.geeksforgeeks.org/applications-of-depth-first-search/)

[DFS](http://www.geeksforgeeks.org/depth-first-traversal-for-a-graph/)

[BFS](http://www.geeksforgeeks.org/breadth-first-traversal-for-a-graph/)

[Detect cycle in a directed graph](http://www.geeksforgeeks.org/detect-cycle-in-a-graph/)

[Find if there is a path b/w two vertices in a directed graph](http://www.geeksforgeeks.org/find-if-there-is-a-path-between-two-vertices-in-a-given-graph/)

[Floyd Warshall Algorithm](http://www.geeksforgeeks.org/dynamic-programming-set-16-floyd-warshall-algorithm/)

[Detect cycle in undirected graph](http://www.geeksforgeeks.org/union-find/)

[Kruskal's Algorithm](http://www.geeksforgeeks.org/greedy-algorithms-set-2-kruskals-minimum-spanning-tree-mst/)

[Graph and its representations](http://www.geeksforgeeks.org/graph-and-its-representations/)

[Prim's algorithm](http://www.geeksforgeeks.org/greedy-algorithms-set-5-prims-minimum-spanning-tree-mst-2/)

[Prim's algorithm 2](http://www.geeksforgeeks.org/greedy-algorithms-set-5-prims-mst-for-adjacency-list-representation/)

[Dijkstra's algorithm](http://www.geeksforgeeks.org/greedy-algorithms-set-6-dijkstras-shortest-path-algorithm/)

[Dijkstra's algorithm 2](http://www.geeksforgeeks.org/greedy-algorithms-set-7-dijkstras-algorithm-for-adjacency-list-representation/)

[Bellman-Ford Algorithm](http://www.geeksforgeeks.org/dynamic-programming-set-23-bellman-ford-algorithm/)

[Transitive closure of a graph](http://www.geeksforgeeks.org/transitive-closure-of-a-graph/)

[Topological sorting](http://www.geeksforgeeks.org/topological-sorting/)

[Shortest path in directed acyclic graph](http://www.geeksforgeeks.org/shortest-path-for-directed-acyclic-graphs/)

[Strongly connected components](http://www.geeksforgeeks.org/strongly-connected-components/)

[Connectivity in directed graph](http://www.geeksforgeeks.org/connectivity-in-a-directed-graph/)

[Detect cycle in an undirected graph 2](http://www.geeksforgeeks.org/detect-cycle-undirected-graph/)

[Applications of BFS](http://www.geeksforgeeks.org/applications-of-breadth-first-traversal/)

**MATRIX**

[Maximum size square submatrix with all 1s](http://www.geeksforgeeks.org/maximum-size-sub-matrix-with-all-1s-in-a-binary-matrix/)

[Turn an image by 90 degree](http://www.geeksforgeeks.org/turn-an-image-by-90-degree/)

[Search in a row wise and column wise sorted matrix](http://www.geeksforgeeks.org/search-in-row-wise-and-column-wise-sorted-matrix/)

[Print a given matrix in spiral form](http://www.geeksforgeeks.org/print-a-given-matrix-in-spiral-form/)

[A boolean matrix question](http://www.geeksforgeeks.org/a-boolean-matrix-question/)

[Min cost path](http://www.geeksforgeeks.org/dynamic-programming-set-6-min-cost-path/)

[Find the row with maximum number of 1s](http://www.geeksforgeeks.org/find-the-row-with-maximum-number-1s/)

[Find the number of islands](http://www.geeksforgeeks.org/find-number-of-islands/)

[Maximum sum rectangle in a 2D matrix](http://www.geeksforgeeks.org/dynamic-programming-set-27-max-sum-rectangle-in-a-2d-matrix/)

[Rotate matrix clockwise](http://www.geeksforgeeks.org/rotate-matrix-elements/)

[Given a boolean matrix. Find k such that all elements in the kth row are 0 and the kth](http://www.geeksforgeeks.org/find-k-such-that-all-elements-in-kth-row-are-0-and-kth-column-are-1-in-a-boolean-matrix/) [column are 1](http://www.geeksforgeeks.org/find-k-such-that-all-elements-in-kth-row-are-0-and-kth-column-are-1-in-a-boolean-matrix/)

[Maximum size rectangle binary submatrix with all 1s](http://www.geeksforgeeks.org/maximum-size-rectangle-binary-sub-matrix-1s/)

**QUEUE**

[Level order traversal](http://www.geeksforgeeks.org/level-order-tree-traversal/)

[Spiral level order traversal](http://www.geeksforgeeks.org/level-order-traversal-in-spiral-form/)

[Implement queue using stacks](http://www.geeksforgeeks.org/queue-using-stacks/)

[Applications of queue](http://www.geeksforgeeks.org/applications-of-queue-data-structure/)

[Implement stack using queues](http://www.geeksforgeeks.org/implement-stack-using-queue/)

[First circular tour that visits all petrol pumps](http://www.geeksforgeeks.org/find-a-tour-that-visits-all-stations/)

[Iterative height of binary tree](http://www.geeksforgeeks.org/iterative-method-to-find-height-of-binary-tree/)

**HEAP**

[k largest elements in an array](http://www.geeksforgeeks.org/k-largestor-smallest-elements-in-an-array/)

[Applications of heap](http://www.geeksforgeeks.org/applications-of-heap-data-structure/)

[Build heap](http://www.geeksforgeeks.org/g-fact-85/)

[Median in a stream of integers](http://www.geeksforgeeks.org/median-of-stream-of-integers-running-integers/)

[Sort a k sorted array](http://www.geeksforgeeks.org/nearly-sorted-algorithm/)

[Sort numbers stored on different machines](http://www.geeksforgeeks.org/sort-numbers-stored-on-different-machines/)

[Merge k sorted arrays](http://www.geeksforgeeks.org/merge-k-sorted-arrays/)

[Print all elements in sorted order from row and column wise sorted matrix](http://www.geeksforgeeks.org/print-elements-sorted-order-row-column-wise-sorted-matrix/)

[kth smallest element in unsorted array](http://www.geeksforgeeks.org/kth-smallestlargest-element-unsorted-array/)

[kth largest element in stream](http://www.geeksforgeeks.org/kth-largest-element-in-a-stream/) -> Zupee

[Why prefer heap over BST for priority queue](http://www.geeksforgeeks.org/why-is-binary-heap-preferred-over-bst-for-priority-queue/)

**HASHING**

[Check for pair in array with sum as x](http://www.geeksforgeeks.org/write-a-c-program-that-given-a-set-a-of-n-numbers-and-another-number-x-determines-whether-or-not-there-exist-two-elements-in-s-whose-sum-is-exactly-x/)

[Vertical sum in binary tree](http://www.geeksforgeeks.org/vertical-sum-in-a-given-binary-tree/)

[Largest subarray with equal number of 0s and 1s](http://www.geeksforgeeks.org/largest-subarray-with-equal-number-of-0s-and-1s/)

[Find if there is a subarray with 0 sum](http://www.geeksforgeeks.org/find-if-there-is-a-subarray-with-0-sum/)

[Print binary tree in vertical order](http://www.geeksforgeeks.org/print-binary-tree-vertical-order-set-2/)

[Special data structure](http://www.geeksforgeeks.org/design-a-data-structure-that-supports-insert-delete-search-and-getrandom-in-constant-time/)

[Find itinerary from a given list of tickets](http://www.geeksforgeeks.org/find-itinerary-from-a-given-list-of-tickets/)

[Largest subarray with 0 sum](http://www.geeksforgeeks.org/find-the-largest-subarray-with-0-sum/)

**BST**

[Find min element](http://www.geeksforgeeks.org/find-the-minimum-element-in-a-binary-search-tree/)

[Check if binary tree is BST](http://www.geeksforgeeks.org/a-program-to-check-if-a-binary-tree-is-bst-or-not/) -> McKinsey

[Inorder successor](http://www.geeksforgeeks.org/inorder-successor-in-binary-search-tree/)

[kth smallest element using order statistics](http://www.geeksforgeeks.org/find-k-th-smallest-element-in-bst-order-statistics-in-bst/)

[Sorted linked list to balanced BST](http://www.geeksforgeeks.org/sorted-linked-list-to-balanced-bst/)

[Construct BST from given preorder traversal](http://www.geeksforgeeks.org/construct-bst-from-given-preorder-traversa/)

[Construct BST from given preorder traversal | Set 2](http://www.geeksforgeeks.org/construct-bst-from-given-preorder-traversal-set-2/)

**PUZZLES**

[Measure 1 litre using 2 vessels and infinite water supply](http://www.geeksforgeeks.org/measure-1-litre-from-two-vessels-infinite-water-supply/)( This problem is not asked to be coded. It is asked only as a puzzle. See [this](http://puzzles.nigelcoldwell.co.uk/twentytwo.htm) too )

[2 eggs 100 floors](http://www.programmerinterview.com/index.php/puzzles/2-eggs-100-floors-puzzle/)

[Mutilated chessboard problem](http://puzzles.nigelcoldwell.co.uk/sixteen.htm)

[100 prisoners, red and blue hats](http://puzzles.nigelcoldwell.co.uk/thirtynine.htm)

[Measure weight of an elephant](http://www.programmerinterview.com/index.php/puzzles/measure-weight-of-an-elephant-with-no-scale/)

[Measure 9 minutes](http://puzzles.nigelcoldwell.co.uk/forty.htm)

[Shortest path in cube](http://puzzles.nigelcoldwell.co.uk/three.htm)

[Angle b/w hour and minute hand](http://puzzles.nigelcoldwell.co.uk/five.htm)

[100 doors puzzle](http://puzzles.nigelcoldwell.co.uk/six.htm)

[Biased to unbiased coin](http://puzzles.nigelcoldwell.co.uk/fiftyseven.htm)

[Red blue pills](http://puzzles.nigelcoldwell.co.uk/fiftyeight.htm)

[25 horses puzzle](http://puzzles.nigelcoldwell.co.uk/fiftynine.htm)

[Poisoned bottles](http://puzzles.nigelcoldwell.co.uk/sixtyone.htm)

[Find the lightest coin](http://www.programmerinterview.com/index.php/puzzles/8-pennies-find-lightest-7-equal/)

[Snail and well problem](http://puzzles.nigelcoldwell.co.uk/sixtytwo.htm)

[Prisoner hat riddle](http://www.programmerinterview.com/index.php/puzzles/hat-puzzle-black-and-white-hats/)

[Cut the cake](http://www.programmerinterview.com/index.php/puzzles/birthday-cake-8-pieces/)

[3 bulbs and switches problem](http://puzzles.nigelcoldwell.co.uk/seven.htm) -> Koo (Indian Twitter)

[Ask the question](http://puzzles.nigelcoldwell.co.uk/fortynine.htm)

[Cheating husbands](http://puzzles.nigelcoldwell.co.uk/nine.htm)

[12 marbles and a scale](http://puzzles.nigelcoldwell.co.uk/one.htm)

[Socks puzzle](http://puzzles.nigelcoldwell.co.uk/fiftytwo.htm)

[Bee and train puzzle](http://puzzles.nigelcoldwell.co.uk/fiftythree.htm)

[Will you die](http://puzzles.nigelcoldwell.co.uk/fiftyfour.htm)

[Globe walker](http://www.mytechinterviews.com/globe-walker)

[Crossing the river](http://puzzles.nigelcoldwell.co.uk/fiftysix.htm)

[Changing your mind](http://puzzles.nigelcoldwell.co.uk/twentythree.htm)

[Divide cards symmetrically](http://puzzles.nigelcoldwell.co.uk/thirtysix.htm)

[Where are you](http://puzzles.nigelcoldwell.co.uk/fortytwo.htm)

[Real and fake coins](http://puzzles.nigelcoldwell.co.uk/fortythree.htm)

[Camel and banana puzzle](http://www.crazyforcode.com/camel-bananas-puzzle/)

[Probability of observing a car](http://puzzles.nigelcoldwell.co.uk/thirtyeight.htm)

[Red and blue marbles](http://puzzlersworld.com/interview-puzzles/red-and-blue-marbles/)

[Warden and 23 prisoners](http://www.techinterviewpuzzles.com/2010/06/warden-and-23-prisoners-google.html)

[Crossing a bridge](http://puzzles.nigelcoldwell.co.uk/twentyfive.htm)

[Age of daughter](http://puzzles.nigelcoldwell.co.uk/thirtyfour.htm)

[Trains and birds](http://www.mytechinterviews.com/trains-and-birds)

[Inverted cards puzzle](http://puzzlersworld.com/interview-puzzles/inverted-cards-puzzle/)

[Aligned clock hands](http://puzzles.nigelcoldwell.co.uk/thirtyfive.htm)

[3 blind men hat color](http://puzzles.nigelcoldwell.co.uk/twelve.htm)

[Gold bar problem](http://www.crazyforcode.com/gold-bar-cuts-puzzle/)

[5 pirates 100 coins](http://www.techinterviewpuzzles.com/2013/07/five-pirates-splitting-100-coins-puzzle.html)

[A box of defective balls](http://www.mytechinterviews.com/one-box-of-defective-balls)

[Probability of having boy](http://puzzlefry.com/puzzles/probability-of-having-boy/)

[Days of month using 2 dice](http://puzzlersworld.com/interview-puzzles/days-of-months-using-2-dice/)

[Red and blue balls](http://puzzlersworld.com/interview-puzzles/red-and-blue-balls/)

[Measure 45 seconds](http://puzzles.nigelcoldwell.co.uk/seventeen.htm) -> McKinsey

[Water level](http://puzzles.nigelcoldwell.co.uk/twentyone.htm)

[8 marbles find heaviest](http://www.programmerinterview.com/index.php/puzzles/8-marbles-puzzle-riddle/)

[100 people with sword](http://puzzlersworld.com/interview-puzzles/100-people-sword-puzzle/)

[Lie tribe and truth tribe](http://puzzlersworld.com/interview-puzzles/lie-tribe-truth-tribe/)

[Monty hall problem](http://betterexplained.com/articles/understanding-the-monty-hall-problem/)

[Girl counting on fingers](http://www.examveda.com/a-young-girl-counted-in-the-following-way-on-the-fingers-of-her-left-hand-she-started-calling-the-thumb-1-the-index-finger-2-middle-finger-3-ring-finger-116/)