**Coding Fundamentals: Perform Each Skill with Each Common Data Structure (if applicable)**

A complete re-arrange means that every element in the array gets moved to a new position. For example, you could move all elements one index forward (and move the last element to the first index) for a complete shift. Or you could invert a binary tree. Or you could rotate a double array. Or you reverse a list, array, queue, etc.

Ideally, you could create a method called

**getRe-ArrangedPosition(int original position)** that calculates the new position of the old position based on the specification of the re-arrange. In addition, ideally you would do this in place.

|  |
| --- |
| **Skill To Master** |
| 1. Traversing forward, backward, and tree traversals. |
| 1. Retrieving the item at the first, last or a particular index |
| 1. Inserting an item at the first, last or a particular index. |
| 1. Removing an item at the first, last or a particular index. |
| 1. Linear Searchig and Binary Searching |
| 1. Basic Sort (Bubble Sort) |
| 1. Complex Sort (i.e. Quick Sort) |
| 1. Complete Re-arrange |

|  |
| --- |
| **Data Structures** |
| Array |
| Double Array |
| Strings |
| Array Lists |
| Linked Lists |
| Stack |
| Queue |
| Binary Tree |
| Binary Search Tree |
| Bit Vector |
| Heaps |