Question: [5*2 marks] Apply both **brute force strategy** as well **as divide & conquer strategy** to solve each given problem. In case any strategy can't be applied on any given part, justify. Also, find **recurrence relation** of each divide & conquer solution.

Note:

- ✓ Your algorithms must be syntax free i.e., they must have the potential to get translated in any high-level language.
- ✓ "It is not mandatory but highly appreciated that you check your attempts via coding your algorithms in your favorite language."
- ✓ Using any built-in language constructs (e.g., python list slicing) for your pseudo-codes are not allowed.
- ✓ You can safely assume that elements are present in our data structures.

Part A: Design an algorithm to add 3 (+3 units) in all even numbers and to add 2 (+2 units) in all odd indices in the first 1/5th elements of the given n integer list.

Part B: Design an algorithm to take sum of 'n' integers given that we want our linear structure (list/array) to be divided into three parts instead of two parts at each recursive step. You need to change the base condition(s) carefully.

Part C: Consider a sorted list which is sorted in ascending-order. This list can only have multiple occurrences of A's, B's and C's. Design an algorithm to count the total number of B's in this sorted list. Make it intelligent by using its sorted order. Note: It is possible that the list has only A's or only B's or only C's. It is also possible that list have no A's or no B's or no C's.

Part D: Consider that you will be given a list of size in powers of 2. (e.g. 2,4,8,16 … etc.) You have to swap the consecutive elements in such a way that first element should be swapped with its immediate neighbor element and so on.

Part E: Consider that you have two arrays of positive integers T and Z. Write a program that compares the content of the two arrays and returns 'true' if the content of both arrays is the same and return 'false' otherwise.

Best of Luck!