# Special Force Lab Assignment Five

**Problem Statement 1:**
You are given an integer array of size n-1, containing distinct integers from 1 to n. Write a Java function to find the missing number in the array. The function should meet the following requirements:

The function should have a time complexity of O(n).
The function should not use any additional memory, i.e., it should be done in-place.
The missing number should be returned as an integer.

**Constraints:**
The size of the input array is in the range [1, 10^6].
The elements of the input array are in the range [1, n], where n is the size of the array.

**Example:**
Input: [1, 2, 4, 5, 6]
Output: 3

**Problem Statement 2:**
You are given an integer array of size n. Write a Java function that prints all even and odd numbers in the array separately. The function should meet the following requirements:

The function should have a time complexity of O(n).
The function should not use any additional memory, i.e., it should be done in-place.
The function should print even and odd numbers separately, in the same order as they appear in the array.

**Constraints:**
The size of the input array is in the range [1, 10^6].
The elements of the input array are in the range [-10^9, 10^9].

**Example:**
Input: [1, 2, 3, 4, 5, 6, 7, 8, 9]
Output: Even numbers: 2 4 6 8; Odd numbers: 1 3 5 7 9

**Problem Statement 3:**
You are given two integer arrays of the same size n. Write a Java function to check if the two arrays are equal or not. The function should meet the following requirements:

The function should have a time complexity of O(n).
The function should not use any additional memory, i.e., it should be done in-place.
The function should return a boolean value indicating whether the two arrays are equal or not.
Constraints:

The size of the input arrays is in the range [1, 10^6].
The elements of the input arrays are in the range [-10^9, 10^9].

# Special Force Lab Assignment Five

**Example:**
Input: [1, 2, 3, 4, 5], [1, 2, 3, 4, 5]
Output: true

**Problem Statement 4:**
Write a Java function that takes in two strings as input and determines whether they are anagrams or not. An anagram is a word or phrase formed by rearranging the letters of a different word or phrase, typically using all the original letters exactly once. The function should meet the following requirements:

The function should ignore whitespace and capitalization while checking for anagrams.
The function should return a boolean value - true if the strings are anagrams, false otherwise.

**Constraints:**
The length of the strings should not exceed 10^5.

**Example:**
Input: "silent", "listen"
Output: true
Input: "rail safety", "fairy tales"
Output: true
Input: "hello", "world
Output: false

**Problem Statement 5:**
Write a Java function that takes in a string containing brackets as input and checks if the brackets are balanced or not. The function should meet the following requirements:

The function should only consider the following brackets - (, ), {, }, [, and ].
The function should return a boolean value - true if the brackets are balanced, false otherwise.

**Constraints:**
The length of the input string should not exceed 10^5.

**Example:**
Input: "({})[]"
Output: true
Input: "(){}[{}]"
Output: true
Input: "([)]"
Output: false

**Instructions : Submit Screenshots of Output of different test cases with the Code. (PDF or Direct)**