1. Given an array, calculate the mode of the array i.e the item that shows up most often. If there are multiple modes, return any one of them.

Example

[1, 1, 1, 2, 7, 1] -> 1

Input: array

Output: int

C++: int CalculateMode(const std::vector<int>& a)  
Java: class Main { public static int calculateMode(List<Integer> a) }  
Python: def calculate\_mode(lst)

1. Given two strings, write a method to decide if one is a permutation of the other.

Example:

'abcd', 'bcda' -> true

'abc', 'bcd' -> false

Input: 2 Strings

Output: Boolean

C++: bool IsPermutation(const std::string& a, const std::string& b)  
Java: class Main { public static boolean isPermutation(String a, String b) }  
Python: def is\_permutation(a, b)

1. Check whether there exists a 1-1 mapping that converts one string to another.  
   Two strings are considered 'similar' if there exists a one-to-one character mapping that makes them equivalent.

Example:

"ABBA" and "ZYYZ" -> true

"ABC" and "GZT" -> true

"ABBA" and "ZYXY" -> false

"ABBA" and "ZYGZ" -> false

Input: 2 Strings

Output: Boolean

C++: bool IsEquivalent(const std::string& a, const std::string& b)  
Java: class Main { public static boolean isEquivalent(String a, String b) }  
Python: def is\_equivalent(a, b)