**Practice Question**

**Function with Array**

**Basic Level**

1. Write a function called findMax that takes an array of integers as input and returns the maximum value in the array.

**Example Input:**  
arr = [3, 1, 4, 1, 5, 9]  
**Output:**  
9

1. Write a function reverseArray that takes an array as input and returns a new array that is the reverse of the original array.

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

1. Create a function isSorted that checks if an array is sorted in ascending order. It should return true if the array is sorted and false otherwise.

**Example Input:**  
arr = [1, 2, 3, 4] → Output: true  
arr = [3, 2, 1] → Output: false

**Intermediate Level**

1. Write a function mergeArrays that takes two sorted arrays as input and returns a single sorted array.

**Example Input:**  
arr1 = [1, 3, 5], arr2 = [2, 4, 6]  
**Output:**  
[1, 2, 3, 4, 5, 6]

1. Write a function findDuplicates that takes an array as input and returns an array of duplicate values (values that appear more than once).

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

**Advanced Level**

1. Write a function rotateArray that rotates an array k positions to the right.

**Example Input:**  
arr = [1, 2, 3, 4, 5], k = 2  
**Output:**  
[4, 5, 1, 2, 3]

1. Implement a function matrixMultiplication that multiplies two 2D arrays (matrices) and returns the resulting matrix.

**Example Input:**

lua

Copy code

mat1 = [[1, 2], [3, 4]]

mat2 = [[2, 0], [1, 2]]

**Output:**

lua

Copy code

[[4, 4],

[10, 8]]

1. Write a function subArraySum that finds all the subarrays of a given array that sum up to a target value.

**Example Input:**  
arr = [1, 2, 3, 4, 5], target = 5  
**Output:**  
[[2, 3], [5]]

**Function**

**Basic Level**

1. **Simple Calculator**  
   Write a function calculate that takes three arguments: two numbers and an operator (+, -, \*, /). The function should return the result of the operation.

**Example Input:**  
calculate(5, 3, '+')  
**Output:**  
8

1. **Factorial**  
   Write a function factorial that takes a number n and returns its factorial.

**Example Input:**  
factorial(5)  
**Output:**  
120

1. **Check Even or Odd**  
   Write a function isEven that takes a number as input and returns true if the number is even, and false otherwise.

**Example Input:**  
isEven(4)  
**Output:**  
true

1. **Power Calculation**  
   Write a function power that takes two arguments: base and exponent, and returns the result of raising the base to the exponent.

**Example Input:**  
power(2, 3)  
**Output:**  
8

**Intermediate Level**

1. **Sum of Digits**  
   Write a function sumOfDigits that takes a number and returns the sum of its digits.

**Example Input:**  
sumOfDigits(123)  
**Output:**  
6

1. **Reverse a Number**  
   Write a function reverseNumber that takes a number and returns it reversed.

**Example Input:**  
reverseNumber(1234)  
**Output:**  
4321

1. **Prime Check**  
   Write a function isPrime that takes a number as input and returns true if the number is prime, and false otherwise.

**Example Input:**  
isPrime(11)  
**Output:**  
true

**Advanced Level**

1. **Greatest Common Divisor (GCD)**  
   Write a function gcd that takes two numbers and returns their greatest common divisor.

**Example Input:**  
gcd(24, 18)  
**Output:**  
6

1. **Fibonacci Sequence**  
   Write a function fibonacci that takes a number n and returns the nth Fibonacci number.

**Example Input:**  
fibonacci(7)  
**Output:**  
13

1. **Armstrong Number**  
   Write a function isArmstrong that checks if a number is an Armstrong number. An Armstrong number is a number equal to the sum of its own digits raised to the power of the number of digits.

**Example Input:**  
isArmstrong(153)  
**Output:**  
true