Object Oriented Programming using Python
Assignment 1

1. Write a program to check whether a number is positive, negative, or zero using if-else.
2. Create a list of integers and find the sum of all even numbers using a loop.
3. Write a function to calculate the factorial of a given number.
4. Implement a Python program to reverse a given string.
5. Create a program to count the frequency of each character in a string.
6. Create a function to check whether a string is a palindrome or not
7. Write a program to find the largest and smallest numbers in a list.
8. Generate a Fibonacci sequence up to n terms using a loop.
9. Write a program to find the sum of digits in a given number.
10. Implement a Python function to check if a number is prime or not.
11. Create a program to split a string into a list of words and display the word count.
12. Implement a program to sort a list of integers in ascending order.
13. Write a Python program to calculate the sum of all numbers in a dictionary.
14. Create a program to count the number of vowels in a given string.
15. Implement a function to check whether a year is a leap year or not.
16. Write a program to remove duplicate elements from a list.
17. Create a program to find the second largest number in a list.
18. Implement a function to find the greatest common divisor (GCD) of two numbers.
19. Write a Python program to calculate the average of numbers in a list.
20. Write a Python program to calculate the sum of the diagonal elements of a matrix.

Object Oriented Programming using Python
Assignment 2

1. Write a program to find whether a number is odd or even using if-else.
2. Create a function to find the length of a list.
3. Write a Python program to count the number of uppercase and lowercase letters in a string.
4. Write a program to merge two dictionaries into one.
5. Implement a function to find the sum of a list of numbers using recursion.
6. Write a program to find all the prime numbers between 1 and n.
7. Write a program to remove all whitespace characters from a string.
8. Implement a function to count the occurrences of a substring in a string.
9. Write a program to find the product of all numbers in a list.
10. Create a program to find the maximum and minimum values in a dictionary.
11. Write a function to check if a string is an anagram of another. An anagram of a string is another string that contains same characters, only the order of characters can be different. For example, "abcd" and "cadb" are anagram of each other.
12. Write a Python program to find the index of the first occurrence of a character in a string.
13. Write a program to replace all occurrences of a character in a string with another character.
14. Create a program to reverse a list without using in-built functions.
15. Write a Python program to check if a string starts with a specific prefix.
16. Implement a function to calculate the power of a number using recursion.
17. Create a program to convert a string to a list of characters.
18. Write a Python program to find the mode of a list of numbers.
19. Create a program to count the frequency of each word in a string.
20. Create a function to split a string at every occurrence of a specific character.

Object Oriented Programming using Python
Assignment 3

1. Write a Python program to check whether a character is a vowel or consonant using if-else.
2. Write a function to find the maximum of three numbers.
3. Create a Python program to merge two sorted lists into one sorted list.
4. Write a program to calculate the sum of even and odd numbers in a list separately.
5. Write a Python program to generate a list of squares of numbers from 1 to n.
6. Implement a function to find the largest element in a 2D matrix.
7. Create a program to calculate the average of digits in a number.
8. Write a Python program to check if a number is an Armstrong number. Armstrong number is a number that is the sum of its own digits raised to the power of the number of digits.
9. Implement a function to remove punctuation from a string.
10. Create a program to sort a list of strings in alphabetical order.
11. Create a program to delete all occurrences of a specific element in a list.
12. Write a program to find the second smallest element in a list.
13. Implement a program to create a dictionary from two lists (keys and values).
14. Write a Python program to check if a given number is divisible by both 3 and 5.
15. Create a program to add elements of two lists element-wise.
16. Implement a function to capitalize the first and last character of each word in a string.
17. Write a program to create a list of tuples with each tuple containing a string and its length.
18. Write a program to calculate the sum of rows and columns in a matrix.
19. Implement a function to find the median of a list of numbers.
20. Write a Python program to calculate the product of all odd numbers in a list.