INTERVIEW QUESTIONS FOR CODING :

Question 1: Write a Python program to check if a string is a palindrome.

Question 2: Write a Python program to find the factorial of a number.

Question 3: Write a Python program to find the largest element in a list.

Question 4: Write a Python program to reverse a string.

Question 5: Write a Python program to count the frequency of each element in a list.

Question 6: Write a Python program to check if a number is prime.

Question 7: Write a Python program to find the common elements between two lists.

Question 8: Write a Python program to sort a list of elements using the bubble sort algorithm.

Question 9: Write a Python program to find the second largest number in a list.

Question 10: Write a Python program to remove duplicates from a list.

Task :

Palindrome

1. Write a Python program to check if a given number is a palindrome.
2. Write a Python program to check if a string is an anagram of a palindrome.

Factorial

1. Write a Python program to find the factorial of a number using a loop.
2. Write a Python program to find the factorial of a number using the `math` library.

Largest Element in a List

1. Write a Python program to find the smallest element in a list.
2. Write a Python program to find the largest even number in a list.

Reverse a String

1. Write a Python program to reverse the words in a sentence.
2. Write a Python program to check if a string is a palindrome without reversing it.

Frequency Count in a List

1. Write a Python program to count the frequency of each character in a string.
2. Write a Python program to count the frequency of each word in a sentence.

Prime Number

1. Write a Python program to check if a number is a perfect number.
2. Write a Python program to check if a number is a strong number.

Common Elements Between Two Lists

1. Write a Python program to find the common elements between two sets.
2. Write a Python program to find the elements that are present in the first list but not in the second.

Bubble Sort

1. Write a Python program to implement the insertion sort algorithm.
2. Write a Python program to sort a list in descending order using bubble sort.

Second Largest Number in a List

1. Write a Python program to find the third largest number in a list.
2. Write a Python program to find the largest odd number in a list.

Remove Duplicates from a List

1. Write a Python program to remove duplicate words from a sentence.

2. Write a Python program to remove duplicates from a list while maintaining the order.

Question 1: Write a Python program to check if a string is a palindrome.

Solution:

```
def is_palindrome(string):
    reversed_string = string[::-1]
    return string == reversed_string

# Test the function
word = "madam"
if is_palindrome(word):
    print(f"{word} is a palindrome")
else:
    print(f"{word} is not a palindrome")
```

Question 2: Write a Python program to find the factorial of a number.

Solution:

```
def factorial(n):
    if n == 0:
        return 1
    else:
        return n * factorial(n - 1)

# Test the function
number = 5
result = factorial(number)
print(f"The factorial of {number} is {result}")
```

Question 3: Write a Python program to find the largest element in a list.

Solution:

```
def find_largest(numbers):
    largest = numbers[0]
    for num in numbers:
        if num > largest:
            largest = num
    return largest

# Test the function
nums = [10, 5, 8, 20, 3]
largest_num = find_largest(nums)
print(f"The largest number is {largest_num}")
```

Question 4: Write a Python program to reverse a string.

Solution:

```
def reverse_string(string):
    return string[::-1]

# Test the function
text = "Hello, World!"
reversed_text = reverse_string(text)
print(reversed_text)
```

Question 5: Write a Python program to count the frequency of each element in a list.

Solution:

```
def count_frequency(numbers):
    frequency = {}
    for num in numbers:
        if num in frequency:
            frequency[num] += 1
        else:
            frequency[num] = 1
    return frequency

# Test the function
nums = [1, 2, 3, 2, 1, 3, 2, 4, 5, 4]
frequency_count = count_frequency(nums)
print(frequency_count)
```

Question 6: Write a Python program to check if a number is prime.

Solution:

```
def is_prime(number):
    if number < 2:
        return False
    for i in range(2, int(number ** 0.5) + 1):
        if number % i == 0:
            return False
    return True

# Test the function
num = 17
if is_prime(num):
```

```
    print(f"{num} is a prime number")
else:
    print(f"{num} is not a prime number")
```

Question 7: Write a Python program to find the common elements between two lists.

Solution:

```
def find_common_elements(list1, list2):
    common_elements = []
    for item in list1:
        if item in list2:
            common_elements.append(item)
    return common_elements

# Test the function
list_a = [1, 2, 3, 4, 5]
list_b = [4, 5, 6, 7, 8]
common = find_common_elements(list_a, list_b)
print(common)
```

Question 8: Write a Python program to sort a list of elements using the bubble sort algorithm.

Solution:

```
def bubble_sort(elements):
    n = len(elements)
    for i in range(n - 1):
        for j in range(n - i - 1):
```

```
            if elements[j] > elements[j + 1]:
                elements[j], elements[j + 1] = elements[j +
1], elements[j]

# Test the function
nums = [5, 2, 8, 1, 9]
bubble_sort(nums)
print(nums)
```

Question 9: Write a Python program to find the second largest number in a list.

Solution:

```
def find_second_largest(numbers):
    largest = float('-inf')
    second_largest = float('-inf')
    for num in numbers:
        if num > largest:
            second_largest = largest
            largest = num
        elif num > second_largest and num != largest:
            second_largest = num
    return second_largest

# Test the function
nums = [10, 5, 8, 20, 3]
second_largest_num = find_second_largest(nums)
print(f"The second largest number is {second_largest_num}")
```

Question 10: Write a Python program to remove duplicates from a list.

Solution:

```
def remove_duplicates(numbers):
    unique_numbers = []
    for num in numbers:
        if num not in unique_numbers:
            unique_numbers.append(num)
    return unique_numbers

# Test the function
nums = [1, 2, 3, 2, 1, 3, 2, 4, 5, 4]
unique_nums = remove_duplicates(nums)
print(unique_nums)
```