Progressive Education Society's

Modern College of Engineering, Pune

MCA Department

A.Y.2023-24

(310908) Python Programming Laboratory

****

Class: FY-MCA Shift / Div : F2 / B Roll Number : 51124

Name: Sameer Kakade Assignment No:3 Date ofImplementation:5/10/2023

****

1. Program to find the sum of all numbers in a list

```
def find_sum(numbers):
   total = 0
   for num in numbers:
      total += num
   return total


numbers = [1, 2, 3, 4, 5]
result = find_sum(numbers)
print(result)
```

Output:

```
   15
```

2. Program to find the largest number in a given list without using max()

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

numbers = [12, 45, 78, 34, 90, 23]
result = find_largest(numbers)
print(result)
```

Output:

```
90
```

3. Program to find the common numbers from two lists

```
def find_common(list1, list2):
    common_numbers = []
    for num in list1:
        if num in list2 and num not in common_numbers:
            common_numbers.append(num)
    return common_numbers

list1 = [1, 2, 3, 4, 5]
list2 = [4, 5, 6, 7, 8]
result = find_common(list1, list2)
print(result)
```

Output:

```
[4, 5]
```

4. Program to print all even numbers from a given list

```
def print_even(numbers):
    for num in numbers:
        if num % 2 == 0:
            print(num, end=' ')

numbers = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
print_even(numbers)
```

Output:

```
    2 4 6 8 10
```

5. Program to create a list of even numbers and another list of odd numbers from a given list

```
def separate_even_odd(numbers):
    even_numbers = []
    odd_numbers = []
    for num in numbers:
        if num % 2 == 0:
            even_numbers.append(num)
        else:
            odd_numbers.append(num)
    return even_numbers, odd_numbers

numbers = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
even, odd = separate_even_odd(numbers)
```

```
print("Even numbers:", even)

print("Odd numbers:", odd)
```

Output:

```
Even numbers: [2, 4, 6, 8, 10]

Odd numbers: [1, 3, 5, 7, 9]
```

6. Program to remove repeated elements from a given list without using built-in methods

```
def remove_duplicates(input_list):

    result = []

    for item in input_list:

        if item not in result:

            result.append(item)

    return result


input_list = [1, 2, 3, 2, 4, 5, 6, 5, 7, 8, 9, 9]

result = remove_duplicates(input_list)

print(result)
```

Output:

```
[1, 2, 3, 4, 5, 6, 7, 8, 9]
```

7. Program to find the longest word in a given sentence

```
def find_longest_word(sentence):

    words = sentence.split()
    longest_word = ""

    for word in words:

        if len(word) > len(longest_word):

            longest_word = word

    return longest_word


sentence = "The quick brown fox jumps over the lazy dog"

result = find_longest_word(sentence)

print(result)
```

Output:

```
 jumps
```

8. Program to find the number of occurrences of a given number without using built-in methods

```
def count_occurrences(numbers, target):

    count = 0

    for num in numbers:

        if num == target:

            count += 1

    return count


numbers = [1, 2, 3, 4, 2, 2, 5, 2, 6]

target = 2

result = count_occurrences(numbers, target)

print(result)
```

Output:

```
4
```

9. Program to print website suffixes (com, org, net, in) from this list

```
def find_suffixes(urls):
    suffixes = ['com', 'org', 'net', 'in']
    website_suffixes = []
    for url in urls:
        parts = url.split('.')
        if len(parts) > 1 and parts[-1] in suffixes:
            website_suffixes.append(parts[-1])
    return website_suffixes

urls = ['www.example.com', 'www.test.org', 'www.google.net', 'www.xyz.in', 'www.invalid']
result = find_suffixes(urls)
print(result)
```

Output:

```
['com', 'org', 'net', 'in']
```

10. Program to sort a given list of numbers without using sort() function

```
def custom_sort(numbers):
    for i in range(len(numbers)):
        for j in range(i+1, len(numbers)):
            if numbers[i] > numbers[j]:
                numbers[i], numbers[j] = numbers[j], numbers[i]

    return numbers


numbers = [3, 1, 4, 1, 5, 9, 2, 6]

result = custom_sort(numbers)

print(result)
```

Output:

```
[1, 1, 2, 3, 4, 5, 6, 9]
```