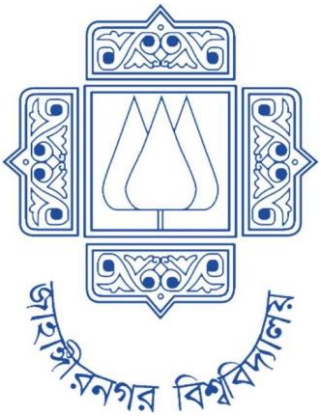
Institute of Information Technology (IIT)

Jahangirnagar University



Lab Report: 03

Submitted by:

Name: Md. Shakil Hossain

Roll No: 2023

Lab Date: 14-June-2023

Submission Date: 22-June-2023

Lab Report: Day 03

Exercise 01:

Problem Name: Write a Python program to find the sum of all the elements in a list.

Code:

```
list=[3,5,4,7,8,1,0,8,4]
total=0
for i in list:
    total+=i
print(f"Sum of list : {total}")
```

Output:

```
Sum of list : 40
```

Exercise 02:

Problem Name: Write a Python program to find the largest, smallest, second largest, and second smallest elements in a list.

Code:

```
list=[3,5,4,7,9,1,0,8,4]
list.sort()
print(f"The list : {list}")
print(f"Largest : {list[-1]}")
print(f"Smallest : {list[0]}")
print(f"Second Largest : {list[-2]}")
print(f"Second Smallest : {list[1]}")
```

Output:

```
The list : [0, 1, 3, 4, 4, 5, 7, 8, 9]
Largest : 9
Smallest : 0
Second Largest : 8
Second Smallest : 1
```

Exercise 03:

Problem Name: Write a Python program to count the number of occurrences of each character in a string.

Code:

```
def charcount(string):
    count={}
    for i in string:
        if i in count:
            count[i]+=1
        else:
            count[i]=1
    return count

string="Beekeepers keep bees"
chars=charcount(string)
for char in chars:
    print(f"{char} :: {chars[char]}")
```

Output:

```
B :: 1
e :: 9
k :: 2
p :: 2
r :: 1
s :: 2
  :: 2
b :: 1
```

Exercise 04:

Problem Name: Write a Python program to create a tuple with elements from a list and print it.

Code:

```
def maketuple(list):
    tup=tuple(list)
    return tup

list=[3,5,2,4,6,7,8,5,34]
tuple=maketuple(list)
print(tuple)
```

Output:

```
(3, 5, 2, 4, 6, 7, 8, 5, 34)
```

Exercise 05:

Problem Name: Write a Python function that takes a list of numbers as input and returns the largest sum of non-adjacent numbers.

Code:

```
def nonadjecntsum(nums):
    if not nums:
        return 0

    n = len(nums)
    if n <= 2:
        return max(nums)

    max_sum = [0] * n
    max_sum[0] = nums[0]
    max_sum[1] = max(nums[0], nums[1])

    for i in range(2, n):
        max_sum[i] = max(max_sum[i-1], max_sum[i-2] + nums[i])

    return max_sum[n-1]

my_list = [1,3,7,2, 4, 6, 2,9, 5]
result = nonadjecntsum(my_list)
print("Largest sum of non-adjacent numbers:", result)
```

Output:

```
Largest sum of non-adjacent numbers: 23
```

Exercise 06:

Problem Name: Write a Python program to remove duplicates from a list and return the resultant list.

Code:

```
def dupremove(list):
    list1=[]
    for i in list:
        if i not in list1:
            list1.append(i)
    print(f"Duplicates removed list : {list1}")

list = [7,4,7,8,2,0,8,1,0,4,5,6,8,2,9]
dupremove(list)
```

Output:

```
Duplicates removed list : [7, 4, 8, 2, 0, 1, 5, 6, 9]
```

Exercise 07:

Problem Name: Write a Python program to find the common elements between two lists and return the resultant list.

Code:

```
def common(list1,list2):
    result=[]
    for i  in  list1:
        if i in list2:
            result.append(i)
    print(f"common elements : {result}")

list1=[1,4,6,8,0,5]
list2=[7,5,1,9,8,4]
common(list1,list2)
```

Output:

```
common elements : [1, 4, 8, 5]
```

Exercise 08:

Problem Name: Write a Python program to find the first n Fibonacci numbers using recursion.

Code:

```
def fibonacci(n):
    if n==0:
        return
    elif n==1:
        return [0]
    elif n==2:
        return [0,1]
    else:
        seq=fibonacci(n-1)
        seq.append(seq[-1]+seq[-2])
        return seq

n=15
fib=fibonacci(n)
print(f"First {n} Fibonacci numbers : {fib}")
```

Output:

```
First 15 Fibonacci numbers : [0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, 233, 377]
```

Exercise 09:

Problem Name: Write a Python function to replace all occurrences of a substring in a string.

Code:

```
string = "Good morning teacher."
substring="teacher"
replace= "students"
rep=string.replace(substring,replace)
print(rep)
```

Output:

```
Good morning students.
```

Exercise 10:

Problem Name: Write a function to add a key-value pair to a dictionary in Python

Code:

```
def  addtodic(dictionary,key,value):
    dictionary[key]=value
    return

dictionary={'name':'Tokee',
            'roll':'2022'}
key="age"
value=23
addtodic(dictionary,key,value)
print(dictionary)
```

Output:

```
{'name': 'Tokee', 'roll': '2022', 'age': 23}
```

Exercise 11:

Problem Name: Write a function to remove a key from a dictionary in Python

Code:

```
def  delete(dictionary,key):
    dictionary.pop(key)
    return

dictionary={'name':'Tokee' , 'roll':'2022' , 'age':23}
key="roll"
delete(dictionary,key)
print(dictionary)
```

Output:

```
{'name': 'Tokee', 'age': 23}
```

Exercise 12:

Problem Name: Write a function to reverse a list of numbers.

Code:

```
def reverse(list):
    return list[::-1]

list=[1,2,3,4,5,6,7]
print(f"Reversed list : {reverse(list)}")
```

Output:

```
...  Reversed list : [7, 6, 5, 4, 3, 2, 1]
```

Exercise 13:

Problem Name: Write a Python program to find and print the key with the maximum value in a dictionary.

Code:

```
def maxkey(dictionary):
    max_value = 0
    max_key = None
    for key, value in dictionary.items():
        if value > max_value:
            max_value = value
            max_key = key
    return max_key

dictionary = {'a': 10, 'b': 30, 'c': 20}
key = maxkey(dictionary)
print(f"Key with maximum value : {key}")
```

Output:

```
...  Key with maximum value : b
```

Exercise 14:

Problem Name: Write a Python program to merge two dictionaries and create a new dictionary.

Code:

```
def marge(dict1, dict2):
    dict={}
    dict.update(dict1)
    dict.update(dict2)
    return dict


dict1 = {'a': 4, 'b': 7, 'c':10}
dict2 = {'x': 3, 'y': 4, 'z':5}
merged_dict = marge(dict1, dict2)
print(f"Merged dictionary : {merged_dict}" )
```

Output:

```
Merged dictionary : {'a': 4, 'b': 7, 'c': 10, 'x': 3, 'y': 4, 'z': 5}
```

Exercise 15:

Problem Name: Given a list of dictionaries, you want to sort them based on a specific key 'age' in each dictionary. Write a lambda function as the key parameter in the sorted() function to achieve this.

Code:

```
people = [
    {'name': 'Alice', 'age': 25},
    {'name': 'Bob', 'age': 30},
    {'name': 'Charlie', 'age': 20}
]
sort=sorted(people,key=lambda  x:x['age'])
print(sort)
```

Output:

```
[{'name': 'Charlie', 'age': 20}, {'name': 'Alice', 'age': 25}, {'name': 'Bob', 'age': 30}]
```