Week - 3: Lists & Tuples

1. i) Write a program to convert a list and tuple into arrays.

Prog
ram:

```
from array import
array list=[1,2,3]
list_array=array("i",lis
t)
print("list_array:",list
_array) tuple=(4,9,0)
tuple_array=array("i",
tuple)
print("tuple_array:",tu
ple_array) output:
list_array:
array('i',[1,2,3])
tuple_array:array('i',[2,9,0])
```

ii) Write a program to find common values between two arrays.

Prog
ram:

```
from array import
*
array1=array('i',[7,
8,9,8])
array2=array('i',[2
3,33,11,8])
common_values=[
] for i in array1:
for j in array2:
       if i==j:
           common_values.append(i)
common_values=set(common_values)
print(common_value
s) output:
 8
```

2. Write a function called gcd that takes parameters a and b and returns their greatest common divisor. **Program:** def gcd(a, b):

```
while b:        a, b = b, a % b
           return a
        num1 = 48 num2 = 18 result =
        gcd(num1, num2) print("GCD of",
        num1, "and", num2, "is:", result)
```

output:

```
GCD of 48 and 18 is: 6
```

3. Write a function called palindrome that takes a string argument and returns True if it is a palindrome and False otherwise. Remember that you can use the built-in function len to check the length of a string. **Program:** def palindrome(string):

```
           len_string=len(string)
        for i in
        range(0,int(len_string/2)):
        if(string[i]==string[len_strin
        g-1]):
        len_string=len_string-1
                 return
        True
        else:
                 return False
        a=input("enter a string
        ")
        c=palindro
        me(a)
        print(c)
```

output:

```
enter a
string
madam
True
```

4. Find mean, median, mode for the given set of numbers in a list.

Program:

```
import statistics as s
```

```
list=[] n=int(input("enter the
number of elements")) for i in
range(0,n):     print("enter the
element")     elements=int(input())
list.append(elements)
print(s.mean(list),"=mean")
print(s.mode(list),"=mode")
print(s.median(list),"=median")
```

output:

```
enter the number of
elements5 enter the element
3
enter the element
5
enter the element
3
enter the element
6
enter the element
2
3.
8
=
m
ea
n
3
=
m
o
d
e
3 =median
```

5. Write a Python program to create a tuple.

Prog
ram:

```
tuple1 = (1, 2,
3, 4, 5) #
Creating an
empty tuple
```

```
tuple2 = ()
# Creating a tuple with a single element tuple3
= (10,)  # Note the comma after the single
element
tuple4 = tuple([6, 7, 8, 9,
10]) tuple5 = ("apple",
3.14, True)

print("Tuple 1:", tuple1)
print("Tuple 2:", tuple2)
print("Tuple 3:", tuple3)
print("Tuple 4:", tuple4)
print("Tuple 5:", tuple5)
```

output:

```
Tuple 1: (1, 2, 3, 4, 5)
Tuple 2: ()
Tuple 3: (10,)
Tuple 4: (6, 7, 8, 9, 10)
Tuple 5: ('apple', 3.14, True)
```

6.Write a Python program to create a tuple with different data types.

Prog
ram:

```
tuple1=("Apple",Tru
e,30) print("Tuple :
",tuple1) output:
Tuple :  ('Apple', True, 30)
```

7.Write a Python program to check whether an element exists within a tuple.

Prog
ram:

```
a=input("enter the element you want to search in
the tuple: ") tuple1=("Dog","cat",True,50,20)
len_tuple1=len(tuple1) for i in
range(0,len_tuple1):     if(a==tuple1[i]):
        print("Element exist within the
tuple") output:
enter the element you want to search in the tuple: Dog
```

Element exist within the tuple