**Assignment- Essentials of Python Programming**

**Name**: Shweta Gupta **PS No.** 99004956

**List 1:**

**1. In a given list of elements, all elements are equal except the one.Write a code to find the odd man out (Stray number)**

var = [1, 1, 1, 1, 1, 1, 1, 2]  
var\_set = list(set(var))  
dict = {var\_set[0]: 0, var\_set[1]: 0}  
for elem in var:  
 dict[elem] += 1  
if dict[var\_set[0]] > 1:  
 print(var\_set[1])  
else:  
 print(var\_set[0])

**2. In a given list of elements, find the elements which is close to its mean**

list\_2 = [1, 3, 5, 2, 6, 7, 2, 5, 98, 14, 15]  
sum = 0  
for elem in list\_2:  
 sum += elem  
mean = sum / len(list\_2)  
print(mean)  
diff = abs(mean - list\_2[0])  
res = list\_2[0]  
for elem in list\_2:  
 if diff > abs(mean - elem):  
 diff = abs(mean - elem)  
 # print()  
 res = elem  
print(res)

**3. Find the average speed of vehicle, given the distance travelled for fixed time intervals, e.g. [ 0, 0.1, 0.25, 0.45, 0.55, 0.7, 0.9, 1.0 ]**

dist = [0, 0.1, 0.25, 0.45, 0.55, 0.7, 0.9, 1.0]  
# Assuming fixed time interval is 1 sec  
time=1  
total\_dist=sum(dist)  
average\_speed=total\_dist/len(dist)\*time  
print("Average Speed: ", average\_speed)

**4. Find the no. of people in a bus, given the data of people onboarding & alighting at each station**

data\_onboarding=[3,4,5,1,0]  
data\_alighting=[0,2,1,3,2]  
print("No. of people in the bus",sum(data\_onboarding)-sum(data\_alighting))

**5. Find the missing number, given the original list and modified one**

lst = [2, 3, 23, 4, 5, 1]  
modified\_lst = [2, 3, 23, 4, 5]  
for element in lst:  
 if element not in modified\_lst:  
 print(element)

**6. Find the difference between two lowest numbers in the list**

list\_6 = [3, 6, 12, 44, 48, 99]  
list\_6.sort(reverse=False)  
result = abs(list\_6[0] - list\_6[1])  
print(result)

**7. In a given list, count no. of elements smaller than their mean**

list\_7 = [72, 5, 104, 8, 39, 34, 12, 34]  
sum = 0  
for elem in list\_7:  
 sum += elem  
mean = sum / len(list\_7)  
# print(mean)  
count = 0  
for elem in list\_7:  
 if elem < mean:  
 count += 1  
print(count)

**List 2:**

**1. Correct the malformed time string, for e.g "5:70:65" to "6:11:05"**

time\_str = "5:70:65"  
list\_time = time\_str.split(":")  
print(list\_time)  
list\_time.reverse()  
sec = int(list\_time[0]) % 60  
mins = (int(list\_time[1]) + (int(list\_time[0]) // 60)) % 60  
hrs = ((int(list\_time[1]) + (int(list\_time[0]) // 60)) // 60) + (int(list\_time[2]))  
result = str(hrs) + ":" + str(mins) + ":" + str(sec)  
print(result)

**2. Correct the malformed date string, for e.g. "45/8/2018" to "14/9/2018"**

str\_date = "45/08/2018"  
list\_date = str\_date.split("/")  
day\_31\_month=[1, 3, 5, 7, 8, 10, 12]  
if int(list\_date[1]) in day\_31\_month:  
 day = int(list\_date[0]) % 31  
 month=int(list\_date[1]) +int(list\_date[0])//31  
elif int(list\_date[1]) in [4, 6, 9, 11]:  
 day = int(list\_date[0]) % 30  
 month = int(list\_date[1])+int(list\_date[0]) // 30  
else:  
 if int(list\_date[2]) % 4 == 0:  
 day = int(list\_date[0]) % 29  
 month = int(list\_date[1])+int(list\_date[0]) // 29  
 day = int(list\_date[0]) % 28  
 month = int(list\_date[1])+int(list\_date[0]) // 28  
month=month %12  
year=int(list\_date[2])+month //12  
modified\_date=str(day)+"/"+str(month)+"/"+str(year)  
print("Modified Date :",modified\_date)

**3. Convert ip address from "a.b.c.d" format into integer and vice versa**

ip\_adrs="192.168.1.2"  
ip\_list=ip\_adrs.split(".")  
result=0  
base\_ip=256  
exp=3  
for num in ip\_list:  
 result+=int(num)\*(base\_ip\*\*exp)  
 exp-=1  
print(result)

**4. Check whether given string is isogram or not**

str = "pinched"  
str\_set = list(set(str))  
dict = {}  
for elem in str\_set:  
 dict[elem] = 0  
for elem in str:  
 dict[elem] += 1  
check = False  
for key, value in dict.items():  
 if dict[key] != 1:  
 print("String is not an isogram")  
 check = True  
 break  
if check == False:  
 print("String is an isogram!")

**5. Given a string, find the mexican wave**

str = "hello"  
list = []  
for i in range(len(str)):  
 list.append(str[:i] + str[i].upper() + str[i + 1:])  
print(list)

**6. Given a number, find the largest number by deleting single digit (order of digits will remain same)**

number = 93041  
num\_dup = number  
check=10  
div=1  
len=0  
while num\_dup!=0:  
 if num\_dup % 10<check:  
 check=num\_dup%10  
 num\_dup//=10  
 len+=1  
num\_temp=number  
for num in range(len):  
 div = div \* 10  
 if check==num\_temp%10:  
 break  
 num\_temp//=10  
result=int((number//div)\*(div/10)+(number%div)%(div/10))  
print(result)

**7. Given a number, find the largest number by shuffling the digits**

number=228701  
lst=[]  
while number!=0:  
 lst.append(number%10)  
 number//=10  
 # len+=1  
lst.sort(reverse=True)  
lst.pop()  
res=0  
for elem in lst:  
 res=elem+res\*10  
print(res)

**8. Compute the word frequency in given message**

sentence="hello world the sun has risen in the world"  
list\_word = sentence.split(" ")  
word\_set = list(set(list\_word))  
dict = {}  
for elem in word\_set:  
 dict[elem] = 0  
for elem in list\_word:  
 dict[elem] += 1  
print(dict)

**9. RGB to Hex conversion and vice versa, e.g. (255,0,255) into 0xFF00FF**

rgb=(255,0,255)  
result="0X"  
for val in rgb:  
 hex\_val=hex(val).upper()  
 if len(hex\_val[2:])<2:  
 result+="0"  
 result+=(hex\_val[2:])  
print(result)

**10. Generate accumulated strings, e.g. abcd ==> A-Bb-Ccc-Dddd**

str="abcde"  
str\_result=""  
print(str[1:])  
str\_result+=str[0].upper()  
for i in range(len(str[1:])):  
 str\_result+="-"+str[i+1].upper()+str[i+1]\*(i+1)  
print(str\_result)