Deshmukh Shubham Kailas
Roll No: 11

Assignment 2

SET A String

```
# 1) Write a python program to check whether the string is Symmetrical or Palindrome.
"""
A string that, when broken into two halves, produces two similar sequences of characters is called a symmetrical string. That is, the division occurs in the middle.
"""
string = (input("Enter String :"))
half = int(len(string) / 2)

if len(string) % 2 == 0: # even
    first_str = string[:half]
    second_str = string[half:]
else: # odd
    first_str = string[:half]
    second_str = string[half+1:]

# symmetric
if first_str == second_str:
    print(string, 'string is symmertical')
else:
    print(string, 'string is not symmertical')

# palindrome
if first_str == second_str[::-1]:
    print(string, 'string is palindrome')
else:
    print(string, 'string is not palindrome')


"""
Enter String :aibohphobia
aibohphobia string is not symmertical
aibohphobia string is palindrome

Enter String :yoyo
yoyo string is symmertical
yoyo string is not palindrome

"""

# 2) Write a python program to Reverse words in a given String
```

```
string = (input("Enter String :"))

words = string.split()

words = list(reversed(words))

print(" ".join(words))

"""
Enter String :hello python
python hello
"""

# 3) Write a python program to remove i'th character from string in different ways

String = input("Enter the string : ")
i = int(input("Enter the index of character to be removed : "))

resetString = ""

for index in range(len(String)):
    if index != i:
        resetString = resetString + String[index]

print("Entered string : " + String)
print("String formed by removing i'th character : " + resetString)

"""
Enter the string : python
Enter the index of character to be removed : 3
Entered string : python
String formed by removing i'th character : pyton
"""

# Write a Python function to find the Max of three numbers.


n1=int(input("Enter first number: "));

n2=int(input("Enter second number: "));

n3=int(input("Enter Third number: "));

def f():

    if(n1>=n2) and (n1>=n3):

        l=n1

    elif(n2>=n1) and (n2>=n3):
```

```
        l=n2

    else:

        l=n3

    print("Largest number among  the three is",l)

f()

"""
Enter first number: 123
Enter second number: 234
Enter Third number: 543
Largest number among  the three is 543
"""
```

SET A Function

```
# 1) Write a Python function to sum all the numbers in a list.


def sum(numbers):
    total = 0
    for x in numbers:
        total += x
    return total


print(sum((8, 2, 3, 0, 7)))


# Write a Python program to reverse a string.
def reverse_string(str):
    str1 = ""   # Declaring empty string to store the reversed string
    for i in str:
        str1 = i + str1
    return str1

string = input('Enter String : ')

print("The original string is : \n",string)
print("The reverse string is : ",reverse_string(string))

"""
The original string is :
 hello python
The reverse string is :  nohtyp olleh
"""
```

SET B String

```
# 1. Write a python program to print even length words in a string.

def printWords(s):

    string = s.split(' ')

    for word in string:

        # if length is even
        if len(word)%2==0:
            print(word)


# Driver Code
string = input("Enter the String :")
printWords(string)

"""
Enter the String :hello python say hello word
python
word
"""

# 2. Write a python program to accept the strings which contains all vowels

myStr = input("Enter the string : ")

# Checking if the string contains all vowels or not
myStr = myStr.lower()
allVowels = set("aeiou")

for char in myStr:
    if char in allVowels:
        allVowels.remove(char)

print("Entered String is ", myStr)
if len(allVowels) == 0:
    print("Accepted \n")
else:
    print("Not Accepted \nThe string does not contain all vowels")

"""
Enter the string : aeiouAEIOU
Entered String is  aeiouaeiou
Accepted

Enter the string : aeibd
Entered String is  aeibd
```

```
Not Accepted
The string does not contain all vowels
"""

# 3. Write a python program to Count the Number of matching characters in a pair of string
def count(s1, s2):
    c = 0  # counter variable
    j = 0
    for i in s1:
        if s2.find(i) > -0 and j == s1.find(i):
            c = c + 1
        j = j + 1
    print("Matching char: ", c)

s1 = input("Enter the string1 : ")
s2 = input("Enter the string2 : ")
count(s1, s2)

"""
Enter the string1 : python program
Enter the string2 : perl program
Matching char:  6
"""
```

SET B Function

```
# 1. Write a Python function that takes a list and returns a new list with unique elements of the first list.

def unique_list(l):
  x = []
  for a in l:
    if a not in x:
      x.append(a)
  return x

print("Unique elements of the first list :",unique_list([1,2,3,2,3,4,4,5]))

"""
Unique elements of the first list : [1, 2, 3, 4, 5]

"""

# 2. Write a Python function that takes a number as a parameter and check the number is prime or not.

# A prime number (or a prime) is a natural number greater than 1 and that has no positive divisors other than 1 and itself.
```

```
def test_prime(n):
    if (n==1):
        return False
    elif (n==2):
        return True;
    else:
        for x in range(2,n):
            if(n % x==0):
                return False
        return True
print(test_prime(5))


# 3. Write a Python function to check whether a number is perfect or not.

def perfect_number(n):
    sum = 0
    for x in range(1, n):
        if n % x == 0:
            sum += x
    return sum == n
print(perfect_number(6))

"""
The first perfect number is 6, because 1, 2, and 3 are its proper positive divisors, and 1 + 2 + 3 = 6. Equivalently, the number 6 is equal to half the sum of all its positive divisors: ( 1 + 2 + 3 + 6 ) / 2 = 6. The next perfect number is 28 = 1 + 2 + 4 + 7 + 14. This is followed by the perfect numbers 496 and 8128.
"""
```

Practise Programs

```
# 1. Write a Python program to append items from a specified list.

from array import *
num_list = [1, 2, 6, -8]
array_num = array('i', [])
print("Items in the list: " + str(num_list))
print("Append items from the list: ")
array_num.fromlist(num_list)
print("Items in the array: "+str(array_num))


# 2. Write a python program Check if a Substring is Present in a Given String

string = input("Enter string:")
sub_str = input("Enter word:")
if string.find(sub_str) == -1:
    print("Substring not found in string!")
else:
```

```
    print("Substring found in string!")

"""
Enter string:Python is programming lang
Enter word:lang
Substring found in string!

Enter string:python program
Enter word:programming
Substring not found in string!

"""

# 3. Write a python program Words Frequency in String Shorthands

string = input("Enter String :")
print("\nEntered String :", string)

word = {key: string.count(key) for key in string.split()}

print("\nWords in the string :")
print(word)

"""
Enter String :python is programming lang and java also programming lang

Entered String : python is programming lang and java also programming lang

Words in the string :
{'python': 1, 'is': 1, 'programming': 2, 'lang': 2, 'and': 1, 'java': 1, 'also': 1}

"""

# 4. Write a python program Convert Snake case to Pascal case

test_str = input("Enter String :")

# printing original string
print("The original string is : " + test_str)

# Convert Snake case to Pascal case
# Using title() + replace()
res = test_str.replace("_", " ").title().replace(" ", "")

# printing result
print("The String after changing case : " + str(res))

"""
Enter String :hello_shubham
The original string is : hello_shubham
```

```
The String after changing case : HelloShubham
"""

# 5. Write a Python function to calculate the factorial of a number (a non-negative integer). The function accepts the number as an argument.

def factorial(n):
    if n == 0:
        return 1
    else:
        return n * factorial(n-1)
n=int(input("Input a number to compute the factiorial : "))
print(factorial(n))

"""
Input a number to compute the factiorial : 5
120
"""

# 6. Write a Python function to check whether a number is in a given range.

def test_range(n):
    if n in range(0, 9):
        print(n, "is in the range")
    else:
        print("The number is outside the given range.")

n=int(input("Input a number whether a number is in a given range : "))
test_range(n)

"""
Input a number whether a number is in a given range : 5
5 is in the range
"""

    # 7. Write a Python function that accepts a string and calculate the number of upper case letters and lower case letters.


def string_test(str):
    d = {"UPPER_CASE": 0, "LOWER_CASE": 0}
    for c in str:
        if c.isupper():
            d["UPPER_CASE"] += 1
        elif c.islower():
            d["LOWER_CASE"] += 1
        else:
            pass
    print("Original String : ", str)
```

```
    print("No. of Upper case characters : ", d["UPPER_CASE"])
    print("No. of Lower case Characters : ", d["LOWER_CASE"])


str = input("Enter String to calculate the number of upper case letters and lower case letters : ")
string_test(str)

"""
Enter String to calculate the number of upper case letters and lower case letters :
Python LanguAGE
Original String :  Python LanguAGE
No. of Upper case characters :  5
No. of Lower case Characters :  9
"""

# 8. Write a Python program to detect the number of local variables declared in a function.

"""
__code__ The code object representing the compiled function body.
co_nlocals is the number of local variables used by the function (including arguments);
"""
def scope():
   a = 17
   b = 9
   c = 2000
   str = 'Python'

print("Number of local varibales available:",scope.__code__.co_nlocals)

"""
Number of local varibales available: 4
"""

# 9. Write a python program to Remove all duplicates from a given string in Python

"""
An OrderedDict is a dictionary subclass that remembers the order that keys were first inserted.

The fromkeys() method returns a dictionary with the specified keys
"""

from collections import OrderedDict
def remove_duplicate(str):
  return "".join(OrderedDict.fromkeys(str))

str = input("Enter String : ")
```

```
print(remove_duplicate(str))

"""
Enter String : python program
python rgam

"""


# 10. Write a Python function that checks whether a passed string is palindrome or not.

def isPalindrome(string):
    return string == string[::-1]

string = input("Enter String : ")
Palindrome = isPalindrome(string)

if Palindrome:
    print("String is Palindrome")
else:
    print("String is not Palindrome")

"""
Enter String : nayan
String is Palindrome

Enter String : python
String is not palindrome
"""




# 11. Write a Python program that accepts a hyphen-separated sequence of words as input and prints the words in a hyphen-separated sequence after sorting them alphabetically.

words=[n for n in input("Enter Words separated by hypen(-):").split('-')]
words.sort()
print("The Words in a hyphen-separated sequence after Sorting : ",'-'.join(words))

"""
Enter Words separated by hypen(-):python-java-php-perl-html-css
The Words in a hyphen-separated sequence after Sorting :  css-html-java-perl-php-python
"""
```