Week 7- Programs on Functions, Recursion, Callback and Closure

Name: Naren Chandrashekhar

SRN: PES2UG20CS216

Section: G

Program 1	Using Functions generate prime numbers using method of Sieve of eratosthenes.
	Program: **def sieve_eratosthenes(n):** **prime_nums = []** **for i in range(2,n+1):** **if i not in prime_nums:** **print(i)** **for j in range(i*i,n+1,i):** **prime_nums.append(j)** **num=int(input("Enter a number to generate prime numbers till "))** **sieve_eratosthenes(num)**
	Output: D:\PES\Semester 1\Computer Science- Python Programming\PythonLab\Week7>python program1.py Enter a number to generate prime numbers till 100 2 3 5 7 11 13 17 19 23 29 31 37 41 43 47 53 59 61 67 71 73 79 83 89 97
Program 2	a) Reverse the given string using recursion.

	b) Solve tower of Hanoi problem. c) Use recursion to raise a number to a given power n.

	Program:
	#a)
	def reverse(string):
	if len(string) == 0:
	return string
	else:
	return reverse(string[1:]) + string[0]
	
	Str = input("Enter a string ")
	print(reverse(Str))
	
	#b)
	def towerofHanoi(n, source, destination, aux):
	if n==1:
	print("move disc 1 from", source, "tower", destination, "tower")
	return
	towerofHanoi (n-1, source, aux, destination)
	print("Move disc ",n,"from",source,"tower",destination,"tower")
	towerofHanoi(n-1,aux,destination,source)
	
	num = int(input("Enter number of discs "))
	towerofHanoi(num,"left","right","middle")
	
	#c)
	def power(n,b):
	if n==0:
	return 1
	else:
	return b*power(n-1,b)
	
	n = int(input("Enter the value of power "))
	b = int(input("Enter the value of base "))
	a = power(n,b)
	print(a)

	Output:

	D:\PES\Semester 1\Computer Science- Python Programming\PythonLab\Week7>python program2.py Enter a string Times Of India newspaper repapswen aidnI fO semiT Enter number of discs 3 move disc 1 from left tower right tower Move disc 2 from left tower middle tower move disc 1 from right tower middle tower Move disc 3 from left tower right tower move disc 1 from middle tower left tower Move disc 2 from middle tower right tower move disc 1 from left tower right tower Enter the value of power 4 Enter the value of base 3 81
Program 3	a) Use callback to find sum ,double and triple the given number.
	Program: **def double(x):** **x = 2*x** **print("The value of the given number, doubled is",x)** **def triple(y):** **y = 3*y** **print("The value of the given number, tripled is",y)** **def find(z,num):** **z(num)** **a = int(input("Enter a given number "))** **find(double,a)** **find(triple,a)**
	D:\PES\Semester 1\Computer Science- Python Programming\PythonLab\Week7>python program3.py Enter a given number 4 The value of the given number, doubled is 8 The value of the given number, tripled is 12 D:\PES\Semester 1\Computer Science- Python Programming\PythonLab\Week7>python program3.py Enter a given number 6 The value of the given number, doubled is 12 The value of the given number, tripled is 18
Program 4	a) Increment a given number by 5 for n number of times using Closure. b) Find Nth root of a given Number By Closure
	Program: **def increment(x,n):** **def increment1():** **print(x+(n*5))** **return increment1** **x = int(input("Enter the given number "))**

	n = int(input("Enter the number of times the number should be incremented by 5 "))
	y = increment(x,n)
	y()
	
	def root(u,v):
	  def nth_root():
	  print((u)**(1/v))
	  return nth_root
	u = int(input("Enter the given number "))
	v = int(input("Enter the root value "))
	w = root(u,v)
	w()

	Output: D:\PES\Semester 1\Computer Science- Python Programming\PythonLab\Week7>python program4.py Enter the given number 11 Enter the number of times the number should be incremented by 5 7 46 Enter the given number 125 Enter the root value 3 5.0
Program 5	Using Concept of decorator find nth fibonnaci number.

	Program
	def func(outer):
	  x = outer()
	  print("nth Fibonnaci number= ",x)
	
	@func
	def outer():
	  a=1
	  b=1
	  n = int(input("Enter n to find nth Fibonnaci number: "))
	  for i in range(2,n):
	  a,b = b,b+a
	  return b
	
	print(outer)

	Output:

```
D:\PES\Semester 1\Computer Science- Python Programming\PythonLab\Week7>python program5.py
Enter n to find nth Fibonnaci number: 7
nth Fibonnaci number=  13
None

D:\PES\Semester 1\Computer Science- Python Programming\PythonLab\Week7>python program5.py
Enter n to find nth Fibonnaci number: 23
nth Fibonnaci number=  28657
None
```

Program 6 — Design a GUI based Calculator using Tkinter.

Program:

```
from tkinter import *
#creating buttons
root = Tk() #creates a window
root.title("Simple Calculator")

e = Entry(root , width = 35, borderwidth = 5)
e.grid(row = 0 , column = 0 , columnspan = 3 , padx = 10, pady = 10)

def click(number):
    current = e.get()
    e.delete(0,END)
    e.insert(0, str(current) + str(number))

def clear():
    e.delete(0,END)

def add():
    f_n = e.get()
    global f_num
    global math
    math = "addition"
    f_num = int(f_n)
    e.delete(0,END)

def sub():
    f_n = e.get()
    global f_num
    global math
    math = "subtraction"
    f_num = int(f_n)
    e.delete(0,END)

def mul():
    f_n = e.get()
```

```
    global f_num
    global math
    math = "multiplication"
    f_num = int(f_n)
    e.delete(0,END)

def div():
    f_n = e.get()
    global f_num
    global math
    math = "division"
    f_num = int(f_n)
    e.delete(0,END)

def equal():
    s_num = int(e.get())
    e.delete(0, END)
    if math == "addition":
        e.insert(0 , f_num + s_num)
    elif math == "subtraction":
        e.insert(0 , f_num - s_num)
    elif math == "multiplication":
        e.insert(0 , f_num * s_num)
    elif math == "division":
        e.insert(0 , f_num / s_num)
#define buttons
bt1 = Button(root, text = "1", padx = 40 , pady = 20, command = lambda : click(1))#callback
bt2 = Button(root, text = "2", padx = 40 , pady = 20, command = lambda : click(2))#callback
bt3 = Button(root, text = "3", padx = 40 , pady = 20, command = lambda : click(3))#callback
bt4 = Button(root, text = "4", padx = 40 , pady = 20, command = lambda : click(4))#callback
bt5 = Button(root, text = "5", padx = 40 , pady = 20, command = lambda : click(5))#callback
bt6 = Button(root, text = "6", padx = 40 , pady = 20, command = lambda : click(6))#callback
bt7 = Button(root, text = "7", padx = 40 , pady = 20, command = lambda : click(7))#callback
bt8 = Button(root, text = "8", padx = 40 , pady = 20, command = lambda : click(8))#callback
bt9 = Button(root, text = "9", padx = 40 , pady = 20, command = lambda : click(9))#callback
bt0 = Button(root, text = "0", padx = 40 , pady = 20, command = lambda : click(0))#callback
btadd = Button(root, text = "+", padx = 39 , pady = 20, command = add)#callback
btsub = Button(root, text = "-", padx = 41 , pady = 20, command = sub)#callback
btmul = Button(root, text = "*", padx = 40 , pady = 20, command = mul)#callback
btdiv = Button(root, text = "/", padx = 41 , pady = 20, command = div)#callback

btEquals = Button(root, text = "=", padx = 90 , pady = 20, command = equal)#callback
btClear = Button(root, text = "Clear", padx = 79 , pady = 20, command = clear)#callback
```

	bt1.grid(row = 3, column = 0) **bt2.grid(row = 3, column = 1)** **bt3.grid(row = 3, column = 2)** **bt4.grid(row = 2, column = 0)** **bt5.grid(row = 2, column = 1)** **bt6.grid(row = 2, column = 2)** **bt7.grid(row = 1, column = 0)** **bt8.grid(row = 1, column = 1)** **bt9.grid(row = 1, column = 2)** **bt0.grid(row = 4, column = 0)** **btadd.grid(row = 5, column = 0)** **btEquals.grid(row = 5, column = 1, columnspan = 2)** **btClear.grid(row = 4, column = 1, columnspan = 2)** **btsub.grid(row = 6, column = 0)** **btmul.grid(row = 6, column = 1)** **btdiv.grid(row = 6, column = 2)** **root.mainloop()**
	Output:

	 Simple Calculator 7 8 9 4 5 6 1 2 3 0 Clear + = - * /