Ex.No.2a SWAPPING

With a temporary variable

```
a=int(input("Enter the value of a:"))

b=int(input("Enter the value of b:"))
print("Value of a is",a)
print("Value of b is",b)
temp=a
a=b
b=temp
print("Value of a is",a)
print("Value of b is",b)
```

Without a temporary variable

```
a=int(input("Enter the value of a:"))
b=int(input("Enter the value of b:"))
print("Value of a is",a)
print("Value of b is",b)
(a,b)=(b,a)
print("Value of a is",a)
print("Value of b is",b)
```

Ex.No.2b DISTANCE BETWEEN TWO POINTS

```
import math
x1=int(input("Enter the value of x1:"))
x2=int(input("Enter the value of x2:"))
y1=int(input("Enter the value of y1:"))
y2=int(input("Enter the value of y2:"))
distance=math.sqrt((x2-x1)**2+(y2-y1)**2)
print("Distance betwen two points is",distance)
```

Ex.No.2c CIRCULATING THE ELEMENTS OF THE LIST

```
a=[1,2,3]
print(a)
j=len(a)-1
while j>0:
    temp=a[j]
    a[j]=a[j-1]
    a[j-1]=temp
    j=j-1
    print(a)
```

Ex.No.3a(i) PRINT N NUMBERS

```
n=int(input("Enter the value of n:"))
i=1
while i<=n:
    print(i)
    i=i+1
```

Ex.No.3a(ii) PRINT N EVEN NUMBERS

```
n=int(input("Enter the value of n:"))
i=2
while i<=n:
    print(i)
    i=i+2
```

Ex.No.3a(iii) PRINT SINE SERIES

```
import math
n=int(input("Enter the value of n:"))
i=0
while i<=n:
    sin=math.sin(i)
    i=i+1
    print(sin)
```

Ex.No.3a(iv) PRINT FIBBONACCI SERIES

```
n=int(input("Enter the value of n:"))
A=0
B=1
C=A+B
print(A)
print(B)
print(C)
while C<n:
    A=B
    B=C
    C=A+B
    print(C)
```

Ex.No.3a(v) PRINT PRIME NUMBER SERIES

```
n1=int(input("Enter the value of n1:"))
n2=int(input("Enter the value of n2:"))
for num in range(n1,n2+1):
    if num>0:
        for i in range(2,num):
            if(num%i)==0:
                break
        else:
                print(num)
```

Ex.NO.3b NUMBER PATTERN

Program-1:

```
n=int(input("Enter the value of n:"))
for i in range(1,n+1):
    for j in range(1,i+1):
        print(i,end="")
    print()
```

Program-2:

```
n=int(input("Enter the value of n:"))
for i in range(1,n+1):
    for j in range(1,i+1):
        print(j,end="")
    print()
```

Program-3:

```
n=int(input("Enter the value of n:"))
for i in range(1,n+1):
    for j in range(i,0,-1):
        print(j,end="")
    print()
```

Program-4:

```
n=int(input("Enter the value of n:"))
for i in range(n,0,-1):
    for j in range(1,i+1):
        print(j,end="")
    print()
```

Ex.NO.3c PYRAMID PATTERN

Program-1:

```
n=int(input("Enter the number of rows"))
for i in range(n):
    for j in range(n-i-1):
        print(" ",end="")
    for j in range(i+1):
        print(j+1,end=" ")
    print()
```

Program-2:

```
n=int(input("Enter the number of rows"))
for i in range(n):
    for j in range(n-i-1):
        print(" ",end="")
    for j in range(i,-1,-1):
        print(i+1,end=" ")
    print()
```

Program-3:

```
n=int(input("Enter the number of rows:"))
for i in range(n-1,-1,-1):
    for j in range(n-i-1):
        print(" ",end="")
    for j in range(i+1):
        print("*",end=" ")
    print()
```

Ex.No.4a LIST IMPLEMENTATION-ITEMS IN LIBRARY

```
libList=["Books","Newspaper","Maps","Documents","e-books"]
choice=0
while True:
   print("Items present in the Library",libList)
   print("List Operations")
   print("1.Append\n 2.Insert\n 3.Modify\n 4.Delete\n 5.Sorting Ascending order\n 6.Sort in Descending order\n 7.Display\n 8.Exit")
   choice=int(input("Enter your choice:"))
#Append
   if choice==1:
      item=input("Enter an item to appended:")
      libList.append(item)
      print("The item has been Appended")
#Insert
   elif choice==2:
      item=input("Enter an item to Insert:")
      pos=int(input("Enter the position to insert:"))
      libList.insert(pos,item)
      print("The item has been inserted")
#Modify
   elif choice==3:
      pos=int(input("Enter the position of item to be modified:"))
      if pos<len(libList):
         Newitem=input("Enter the Newitem to modify:")
         olditem=libList[pos]
         libList[pos]=Newitem
         print("The item",olditem,"has been modified")
      else:
         print("Given position is incorrect")
#Delete
   elif choice==4:
      item=input("Enter an item to Delete:")
      if item in libList:
         libList.remove(item)
         print("The item has been deleted.")
      else:
         print("The item is not in the list")
#Sort in Ascending order
   elif choice==5:
      libList.sort()
      print("The list has been sorted")
#Sort in Descending order
   elif choice==6:
      libList.sort(reverse=True)
      print("The list hasbeen sorted in reverse order")
#Display
   elif choice==7:
      print("The items present in the Lirary",libList)
#Exit
   elif chocie==8:
      break
   else:
      print("Choice is not valid")
      print("Press any key to continue...")
      ch=input()
```

Ex.No.4b TUPLE IMPLEMENTATION-ITEMS IN LIBRARY

```
libTuple=('Books',"Newspaper","Maps","Documents",'CDs')
print("\nItems present in Library\n",libTuple)
print("\n Tuple Operations\n 1.Add an element\n 2.Update or Delete an element\n 3.Length of a Tuple\n 4.Search an Item\n 5.Sort Tuple elements\n 6.Display\n 7.Delete Tuple\n 8.Exit")
choice=0
while True:
   choice=int(input("Enter your choice:"))
#Add
   if choice==1:
      item=input("Enter an item to add:")
      libTuple=libTuple+(item,)
      print("The item is added")
#Update/Delete an Item-Immutable
   elif choice==2:
      print("Tuple is immutable.\n So the elements cannot be updated or deleted")
#Length
   elif choice==3:
      print("The number of items in the Library:",len(libTuple))
#Search
   elif choice==4:
      item=input("Enter an item to search:")
      if item in libTuple:
         print(item,"is present in the position",libTuple.index(item)+1)
#Sort
   elif choice==5:
      print("The sorted Items:\n",sorted(libTuple))
#Display
   elif choice==6:
      print("The Items in the Library:\n",libTuple)
#Delete Tuple
   elif choice==7:
      del libTuple
      print("The tuple is deleted.")
#Exit
   elif choice==8:
      break
   else:
      print("Choice is not valid")
      print("\nPress any key to continue...\n")
      ch=input()
```

Ex.No.5a CAR COMPONENTS-DICTIONARY

```
d={1:'engine',2:'wheel',3:'gear',5:'break',4:'steering'}
print("\ncomponents present in the car\n",d)
print("\nDictionary Operations\n 1.Adding/Updating\n 2.Deleting elements\n 3.Length of Dictionary\n 4.Display keys\n 5.Display Items\n 6.Sorting\n 7.Delete Dictionary\n 8.Exit")
choice=0
while True:
   choice=int(input("Enter your choice:"))
#Adding/Updating
   if choice==1:
      key=int(input("Enter your key:"))
      value=input("Enter your value:")
      d[key]=value
      print("The value is added or updated")
#Deleting elements
   elif choice==2:
      del d[2]
      print(" The parts has been deleted")
#Length of Dictionary
   elif choice==3:
      print("The number of components present in the car:",len(d))
#Display keys
   elif choice==4:
      print("The keys present in the dictionry",d.keys())
#Display Items
   elif choice==5:
      print("The parts present in the dictionry",d.values())
#Sorting
   elif choice==6:
      print("The sorted dictionary:\n",sorted(d))
#Delete dictionary
   elif choice==7:
      del d
      print("The dictionary has been deleted.")
#Exit
```

```
    elif choice==8:
        break
    else:
        print("Choice is not valid")
      print("\nPress any key to continue...\n")
        ch=input()
```

Ex.No.5b AUTOMOBILE COMPONENTS-SET

```
CarSet={"Engine","Battery","Breaks","Transmission"}
print("\ncomponents of an Automobile\n",CarSet)
print("\nSet Operations\n 1.Add a component\n
2.Union\n 3.Intersection\n 4.Difference\n 5.Length of
the Set\n 6.Copy of the Set\n 7.Clear\n 8.Search\n
9.Display\n 10.Delete\n 11.FrozenSet\n 12.Exit")
choice=0
while True:
    choice=int(input("Enter your choice:"))
#Add
    if choice==1:
        comp=input("Enter a component to add:")
        CarSet.add(comp)
        print("The component is added in the set")
#Union
    elif choice==2:
        CarSetnew={"Radiator","Alternator"}
        CarSet=CarSet.union(CarSetnew)
        print(" The set After Union Operation:\n",CarSet)
#InterSection
    elif choice==3:
        CarSetnew={"Battery","Front_Axile","Engine"}
        CarSet=CarSet&CarSetnew
        print("The set After Intersection
operation:\n",CarSet)
#Difference
    elif choice==4:

CarSetnew={"Engine","Battery","Brakes","Transmissio
n"}
        CarSet=CarSetnew-CarSet
        print("The Set After Difference
operation:\n",CarSet)
#Length
    elif choice==5:
        print("The number of components of an
automobile in the Set:\n",len(CarSet))
#Copy
    elif choice==6:
        CopySet=CarSet.copy()
        print("The new copy of the Set:\n",CopySet)
#Clear
    elif choice==7:
        CopySet.clear()
        print("The CopySet is cleared.")
#Search
    elif choice==8:
        comp=input("Enter a component to search:")
        if comp in CarSet:
            print(comp,"is preent in the set")
        elif comp not in CarSet:
            print(comp,"is not in the present in the set")
#Display
    elif choice==9:
        print("The components of an Automobile present
in the set:\n",CarSet)
#Delete
    elif choice==10:
        comp=input("Enter a component to delete:")
        CarSet.discard(comp)
        print("The component is deleted")
#frozenset()
    elif choice==11:
        CarSet=frozenset(CarSet)
        print("After frozenset(),the set will be Immutable")
#Exit
    elif choice==12:
        break
    else:
        print("Choice is not valid")
        print("\nPress any key to continue...\n")
        ch=input()
```

Ex.No.6a FACTORIAL

```
def fact(n):
    if n==0:
        return 1
    else:
        return n*fact(n-1)
n=int(input("Enter the value of n:"))
print("The factorial is",fact(n))
```

Ex.No.6b LARGEST NUMBER IN A LIST

```
def large(a):
    s=a[0]
    for i in range(0,len(a)):
        if s<a[i]:
            s=a[i]
    print("Largest number in a list",s)
a=[7,3,0,9,8]
large(a)
```

Ex.No.6c AREA OF SHAPE

```
def square():
    a=int(input("Enter a:"))
    area=a*a
    print("Area of square is",area)
def circle():
    r=int(input("Enter r:"))
    area=3.14*r*r
    print("Area of circle is",area)
def rectangular():
    l=int(input("Enter l:"))
    b=int(input("Enter b:"))
    area=l*b
    print("Area of rectangular is",area)
def triangle():
    b=int(input("Enter b:"))
    h=int(input("Enter h:"))
    area=b*h/2
    print("Area of triangle is",area)
def semicircle():
    r=int(input("Enter r:"))
    area=3.14*r*r/2
    print("Area of equation is",area)
print("\nArea of shape\n 1.square\n 2.circle\n
3.rectangular\n 4.triangle\n 5.semicirle\n 6.Exit")
choice=0
while True:
    choice=int(input("Enter your choice:"))
#square
    if choice==1:
        square()
#circle
    elif choice==2:
        circle()
#rectangular
    elif choice==3:
        rectangular()
#triangle
    elif choice==4:
        triangle()
#semicircle
    elif choice==5:
        semicircle()
#Exit
    elif choice==6:
        break
    else:
        print("Choice is not valid")
        print("Press any key to continue...")
        ch=input()
```

EX.NO.7A STRING REVERSE

```
def strrev(string):
    str=""
    for i in string:
        str=i+str
    print("Reversed string:",str)
string=input("Enter your string:")
strrev(string)
```

Ex.No.7b STRING PALINDROME

```
def strrev(string):
    str=""
    for i in string:
        str=i+str
    print("Reversed string:",str)
    if(str==original):
        print("This is palindrome")
    else:
        print("This is not palindrome")
string=input("Enter your string:")
original=string
str=0
strrev(string)
```

Ex.No.7c CHARACTER COUNT

```
string=input("Enter your string:")
c=input("Enter your character to check frequency:")
count=0
for i in string:
    if i==c:
        count+=1
print(c,"occurs",count,"time(string).")
```

Ex.No.7d REPLACING CHARACTERS

```
str1=input("Enter your string:")
ch=input("Enter your character:")
newch=input("Enter your newcharacter:")
str2=str1.replace(ch,newch)
print("Original string:",str1)
print("Modified string;",str2)
```

EX.NO.8 a PANDAS

```
import pandas as pd
data={“Country”:[“Brazil”,  “Russia”, “India”, “China”, “South Africa”], “Capital”:[“Brasilia”, “Moscow”, “New Delhi”, “Beijing”, “Pretoria”], “area”: [8.516,17.10,3.286,9.597,1.221], “Population”: [200.4,143.5,1252,1357,52.98]
data_table=pd.DataFrame(data)
print(data_table)
```

EX.NO.8b NUMPY

```
import numpy as np
x=np.array([[1,2],[3,4]])
y=np.array([[5,6],[7,8]])
v=np.array([9,10])
w=np.array([11,12])
print(np.dot(v,w), “\n”)
print(np.dot(x,v), “\n”)
print(np.dot(x,y))
```

EX.NO.8c MATPLOTLIB

```
#importing matplotlib module
from matplotlib import pyplot as plt
#x-axis values
x=[5,2,9,4,7]
#y-axis values
y=[10,5,8,4,2]
#Function to plot
plt.plot(x,y)
 #Function to show the plot
plt.show()
```

EX.NO.8d SCIPY

```
from Scipy.misc import imread,imsave,imresize
img=imread(‘ball.jpeg’)
print(img.dtype,img.shape)
img_tint=img *  [1,0.45,0.3]
imsave(‘ball-tinted.jpeg’,img_tint)
img_tint_resize=imresize(img_tint,(300,300))
imsave(‘ball-tinted-resized.jpeg’,img_tint_resize)
```

EX.NO.9a COPY FROM ONE FILE TO ANOTHER

```
fs=open(“sample.txt”, “r+”)
fs.write(“Hello python”)
content=fs.read()
fd=open(“d.txt”, “w+”)
try:
        fd.write(content)
        print(“file copied”)
except
        print(“unable to copy”)
fd.close()
fd.close()
```

EX.NO.9b WORD COUNT

```
Print(“Printing and count in text file”)
File 1=open(“Sample.txt”, “r+”)
word count={}
for word in file 1.read().split():
        if word not in wordcount:
                wordcount[word]=1
        else
                wordcount[word]=+1
sum=0
for k,v in wordcount.items():
        print(str(k)+  “ – “ +str(v))
        sum=sum+v
print(“Total number of words in file:”,sum)
file 1.close()
```

EX.NO.9c LONGEST WORD

```
file 1=open(“sample.txt”, “r”)
str=file 1.read()
words=str.split()
max_len=len(max(words,key=len))
for words in words:
        if(len(word)==max_len):
                longest_word=word
print(“The longest word in file is:”, longest_word)
```

EX.NO.10a DIVIDE BY ZERO ERROR

```
a=int(input(“Enter a number:”))
b=int(input(“Enter a dividing number:”))
try
        if(b==0):
                print(“You cannot divide by zero!”)
        else
                print(“The Division is:”,a/b)
except ZeroDivisonError:
        print(“Enter a valid number”)
finally:
        print(“Thankyou”)
```

EX.NO.10b VOTER’S AGE VALIDITY

```
age=int(input(“Enter your age:”))
try:
        if(age>18):
                print(“Eligible to vote”)
        else
                print(“Not  Eligible to vote”)
except:
        print(“Enter a valid age”)
finally:
        print(“Thankyou”)
```

EX.NO.10c STUDENT MARK VALIDATION

```
mark=int(input(“Enter your mark:”))
try:
        if(mark>=0):
                print(“Your mark is:”,mark)
```

```
        else:
                print(“Enter a mark between 0 to 100”)
except:
        print(“Enter a valid input”)
finally:
        print(“Thankyou”)
```

EX.NO.12 PYGAME

```
import pygame
import time
pygame.init()
screen=pygame.display.set_mode((500,300))
y=1
direction=1
counter=0
while True:
        screen.fill((255,255,255))
        pygame.draw.circle(screen,(0,255,0),(250,y)1
3,0)
        pygame.display.update()
        time.sleep(.008)
        if  y==300:
                direction=-1
        elif  y==0:
                direction=1
                counter=counter+1
        y=y+direction
        if counter==3:
                pygame.quit()
                break
```