EXP-1

num1=int(input("Enter the first number:-"))

num2=int(input("Enter the second number:-"))

while num2!=0:

(num1,num2)=(num2,num1%num2)

print("The GCD of the given two numbers are ",num1)

EXP-2

num=int(input("Enter the number "))

approx=(num/2)

for i in range(10):

result=0.5\*(approx+(num/approx))

approx=result

print("The square root of",num,"=",approx)

EXP-3

num1=int(input("Enter the number 1:-"))

num2=int(input("Enter the number 2:-"))

p=1

if(num2>0):

for i in range(num2):

p=p\*num1

else:

p=p\*x

p=(1/p)

print(num1,"raised to ",num2,"=",p)

EXP-4

n = int(input("Enter the length of the array: "))

array = []

for i in range(n):

l = int(input(f"Enter Element {i}: "))

array.append(l)

print("The array elements are: ")

print(array)

for i in range(n):

for j in range(i, n):

if array[i] < array[j]:

temp = array[i]

array[i] = array[j]

array[j] = temp

print("Max 1 =", array[0])

print("Max 2 =", array[1])

EXP-5

LINEAR SEARCH  
  
no\_of\_elements = int(input('Enter the size of the list: '))

array = []

print('Enter the elements of the list: ')

for i in range(0, no\_of\_elements, 1):

element = int(input())

array.append(element)

search\_element = int(input('Enter the search element: '))

flag = 0

for i in range(0, no\_of\_elements, 1):

if search\_element == array[i]:

flag = 1

break

if flag == 1:

print('Element is found at index: {}'.format(i))

else:

print('Element not found')

B)BINARY SEARCH  
  
def binary\_search(arr, low, high, x):

if high >= low:

mid = (high + low) // 2

if arr[mid] == x:

return mid

elif arr[mid] > x:

return binary\_search(arr, low, mid - 1, x)

else:

return binary\_search(arr, mid + 1, high, x)

else:

return -1

arr\_input = input("Enter the array elements separated by spaces: ").split()

arr = [int(num) for num in arr\_input]

x = int(input("Enter the element to search for: "))

result = binary\_search(arr, 0, len(arr) - 1, x)

if result != -1:

print("Element is present at index", result)

else:

print("Element is not present in the array")

EXP-6

SELECTION SORT  
  
def selectionSort(arr, size):

for ind in range(size):

min\_index = ind

for j in range(ind + 1, size):

if arr[j] < arr[min\_index]:

min\_index = j

(arr[ind], arr[min\_index]) = (arr[min\_index], arr[ind])

size = int(input("Enter the size of the array: "))

arr = []

for i in range(size):

arr.append(int(input(f"Enter Element {i+1}: ")))

selectionSort(arr, size)

print('The array after sorting in ascending order by selection sort is:')

print(arr)  
  
  
  
B) INSERTION SORT   
  
def insertionSort(arr, size):

for i in range(1, size):

key = arr[i]

j = i - 1

while j >= 0 and key < arr[j]:

arr[j + 1] = arr[j]

j -= 1

arr[j + 1] = key

size = int(input("Enter the size of the array: "))

arr = []

for i in range(size):

arr.append(int(input(f"Enter Element {i+1}: ")))

insertionSort(arr, size)

print('The array after sorting in ascending order by Insertion sort is:')

print(arr)  
  
  
  
  
  
C) MERGE SORT  
  
def mergeSort(array):

if len(array) > 1:

r = len(array) // 2

L = array[:r]

M = array[r:]

mergeSort(L)

mergeSort(M)

i = j = k = 0

while i < len(L) and j < len(M):

if L[i] < M[j]:

array[k] = L[i]

i += 1

else:

array[k] = M[j]

j += 1

k += 1

while i < len(L):

array[k] = L[i]

i += 1

k += 1

while j < len(M):

array[k] = M[j]

j += 1

k += 1

array = []

n = int(input("Enter the size of the array: "))

for i in range(n):

array.append(int(input(f"Enter Element {i+1}: ")))

mergeSort(array)

print("Sorted array is:")

print(array)  
  
EXP-7

def matrix\_multiplication(matrix1, matrix2):

result = []

for \_ in range(len(matrix1)):

row = []

for \_ in range(len(matrix2[0])):

row.append(0)

result.append(row)

for i in range(len(matrix1)):

for j in range(len(matrix2[0])):

for k in range(len(matrix2)):

result[i][j] += matrix1[i][k] \* matrix2[k][j]

return result

def get\_matrix\_from\_user(prompt):

print(prompt)

rows = int(input("Enter the number of rows: "))

cols = int(input("Enter the number of columns: "))

matrix = []

for i in range(rows):

row = []

for j in range(cols):

element = int(input(f"Enter element at row {i+1}, column {j+1}: "))

row.append(element)

matrix.append(row)

return matrix

matrix1 = get\_matrix\_from\_user("Enter the first matrix:")

matrix2 = get\_matrix\_from\_user("Enter the second matrix:")

result = matrix\_multiplication(matrix1, matrix2)

print("Result: ")

for row in result:

for col in row:

print(col, end=" ")

print()  
  
  
  
  
  
EXP 8  
  
import sys

program\_name = sys.argv[0]

list\_1 = []

for i in range(1, len(sys.argv)):

argument = sys.argv[i]

list\_1.append(argument)

print("Word count:", len(sys.argv) - 1)

print("Unsorted list:")

print(\*list\_1, sep="\n")

list\_1.sort()

print("Sorted list:")

print(\*list\_1, sep="\n")