Write a program to determine largest command line input

from sys import argv

try:

    a=[]

    for i in range(1,len(argv)):

        a.append(eval(argv[i]))

    print(max(a))

except NameError:

    print("give string input in single or triple quotes ")

Write a program to determine command line input is even number or odd number

from sys import argv

try:

    if (int(argv[1])) %2==0:

        print("Even")

    else:

        print("Odd")

except NameError:

    print("input must be integer")

except :

    print("enter any integer ")

Write a program to determine average of command line inputs

from sys  import argv

try:

    a=[]

    for i in range(1,len(argv)):

        a.append(eval(argv[i]))

    s=sum(a)

    b=len(a)

    print(s/b)

except NameError:

    print("Pls send number input")

except ZeroDivisionError:

    print(" Pls  send  number  inputs")

Write a program to sort command line inputs in ascending order and descending order

from sys import argv

try:

    a=[]

    for i in range(1,len(argv)):

        a.append(eval(argv[i]))

    b=sorted(a)

    c=sorted(a,reverse=True)

    print("sorted array",b)

    print("desc sorted array",c)

except NameError:

    print("enter str in single or triple quotes")

# Find outputs (Home work)

print( 'green' in 'Hyd is green city') #True

print('day' in 'Sankar dayal sarma') #True

print('Green' in 'Hyd is green city') #False

print('d is' in 'Hyd is green city') #True

print('dis' in 'Hyd is green city') #False

print('iniv' in 'Srinivas') #True

print('iniv' not in   'Srinivas') #False

''' (Home work)

Slice demo program

0 1 2 3 4 5 6 7

R a m a R a o

-8 -7 -6 -5 -4 -3 -2 -1

'''

a = 'Rama Rao'

print(a [ : 7 : 2]) #Rm a

print(a [ : 7]) #Rama R

print(a [2 : 4]) #ma

print(a [2 : ]) #ma Rao

print(a [ : 4 ]) # Rama

print(a [ : : 2]) #Rm a

print(a [-6 : -1]) # ma Ra

print(a [-6 : ]) # ma Rao

print(a [: -4 : -1]) #oaR

print(a [-3 : -1]) # a[-3 : -1 : 1] ---> string from indexes -3 to -2 in steps of 1 i.e. Ra

print(a [-3 : ]) #Rao

print(a [ : : ]) #Rama Rao

print(a [ : ]) #Rama Rao

print(a [ : : -1]) #oaR amaR

print(a [ : : -2]) #oRaa

print(a [ -2 : : -2]) # a[-2 : -9 : -2] ---> string from indexes -2 to -8 in steps of -2 i.e. a<space>mR

print(a [2 : 8]) #ma Rao

print(a [2 : 8 : -1]) #

print(a [ : -6 : -1]) #oaR a

print(a [2 : -3]) #ma

print(a [1 : 6 : 2]) #aaR

print(a [ : -5 : -5]) #o

print(a [2 : -5]) #m

print(a [2 : -5 : 2]) #m

print(a [ : 0 : -1]) #oaR ama

print(a [-5 : 0 : -2]) #aa

# len() function demo program (Home work)

print(len('Hyd')) # 3

print(len('Rama Rao'))#8

print(len('9247'))#4

print(len('+-$'))#3

print(len(''))#0

print(len(' '))#1

print(len('A2#'))#3

print(len(3456)) #error int has no length

print('Sec'. len()) #error len(‘Sec’)

# chr() function demo program

print(chr(65)) # Converts unicode value 65 to 'A'

print(chr(90)) #Z

print(chr(97))#a

print(chr(122))#z

print(chr(48))#0

print(chr(57))#9

print(chr(36))#$

print(chr(32))#<space>

print(ord('A')) # Converts 'A' to unicode value 65

print(ord('Z')) #90

print(ord('a'))#97

print(ord('z'))#122

print(ord('0'))#48

print(ord('9'))#57

print(ord('$'))#36

print(ord(' '))#32