[Practica 2]

[Sistema de Visión Artificial]

Loera Contreras Gildardo Yair – 19110230 – 7E

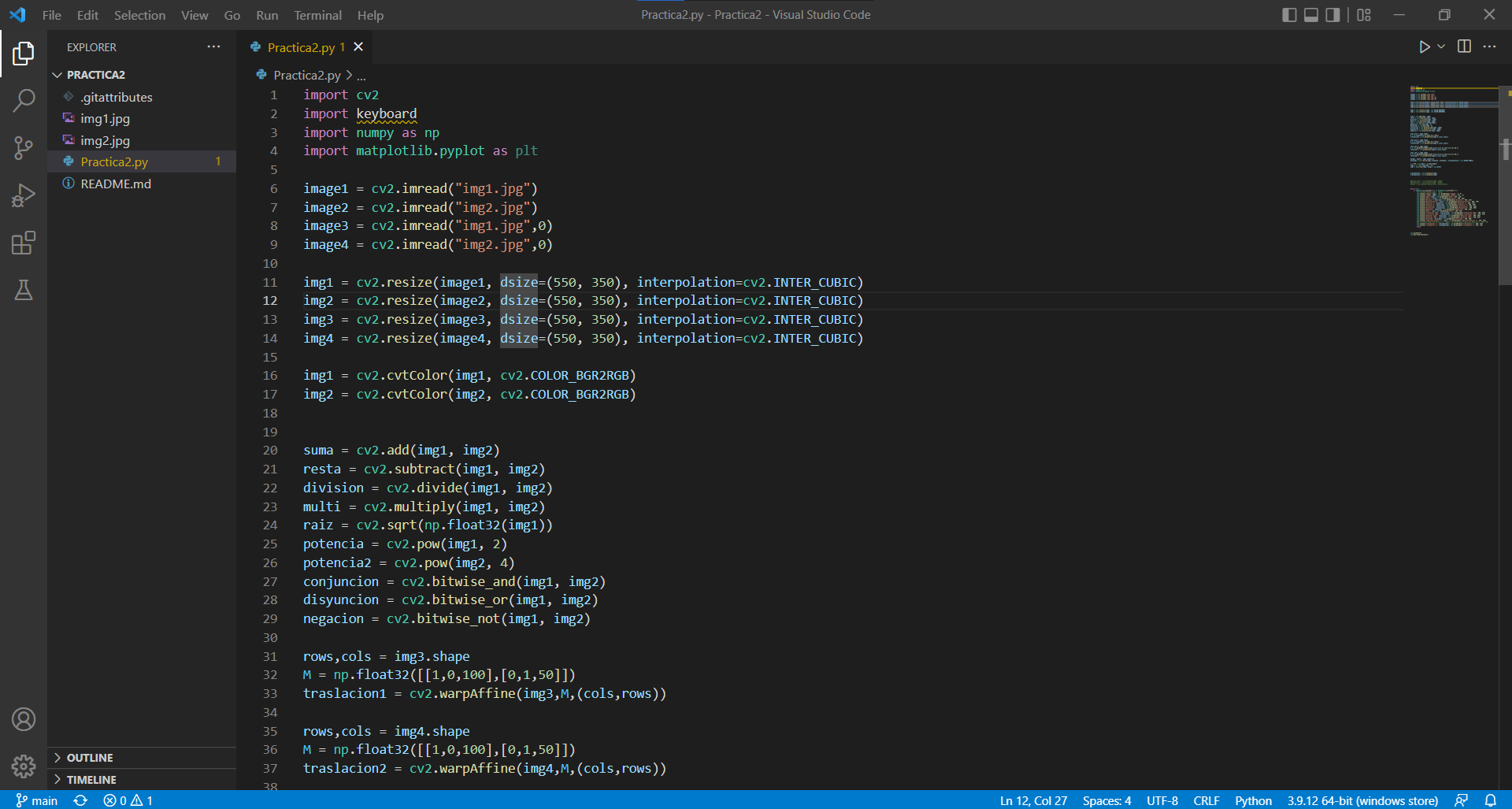
profesor [Mauricio alejandro cabrera arellano]



**PROBLEMA**

Mostrar 2 imágenes al pulsar La primera letra de su nombre las imágenes se pondrán a los lados y a centro aparecerá:  
La suma  
La resta  
La división  
La Multiplicación  
Logaritmo natural  
Raíz  
Derivada  
Potencia  
Conjunción  
Disyunción  
Negación  
Traslación  
Escalado  
Rotación  
Traslación A fin.  
Transpuesta  
Proyección

**CODIGO**



import cv2

import keyboard

import numpy as np

import matplotlib.pyplot as plt

image1 = cv2.imread("img1.jpg")

image2 = cv2.imread("img2.jpg")

image3 = cv2.imread("img1.jpg",0)

image4 = cv2.imread("img2.jpg",0)

img1 = cv2.resize(image1, dsize=(550, 350), interpolation=cv2.INTER\_CUBIC)

img2 = cv2.resize(image2, dsize=(550, 350), interpolation=cv2.INTER\_CUBIC)

img3 = cv2.resize(image3, dsize=(550, 350), interpolation=cv2.INTER\_CUBIC)

img4 = cv2.resize(image4, dsize=(550, 350), interpolation=cv2.INTER\_CUBIC)

img1 = cv2.cvtColor(img1, cv2.COLOR\_BGR2RGB)

img2 = cv2.cvtColor(img2, cv2.COLOR\_BGR2RGB)

suma = cv2.add(img1, img2)

resta = cv2.subtract(img1, img2)

division = cv2.divide(img1, img2)

multi = cv2.multiply(img1, img2)

raiz = cv2.sqrt(np.float32(img1))

potencia = cv2.pow(img1, 2)

potencia2 = cv2.pow(img2, 4)

conjuncion = cv2.bitwise\_and(img1, img2)

disyuncion = cv2.bitwise\_or(img1, img2)

negacion = cv2.bitwise\_not(img1, img2)

rows,cols = img3.shape

M = np.float32([[1,0,100],[0,1,50]])

traslacion1 = cv2.warpAffine(img3,M,(cols,rows))

rows,cols = img4.shape

M = np.float32([[1,0,100],[0,1,50]])

traslacion2 = cv2.warpAffine(img4,M,(cols,rows))

rows,cols = img3.shape

M = cv2.getRotationMatrix2D(((cols-1)/2.0,(rows-1)/2.0),90,1)

rotacion1 = cv2.warpAffine(img3,M,(cols,rows))

rows,cols = img4.shape

M = cv2.getRotationMatrix2D(((cols-1)/2.0,(rows-1)/2.0),90,1)

rotacion2 = cv2.warpAffine(img4,M,(cols,rows))

height, width = img1.shape[:2]

escalado = cv2.resize(img1,(2\*width, 2\*height), interpolation = cv2.INTER\_CUBIC)

c = 255 / np.log(1 + np.max(img1))

logn = c \* (np.log(img1 + 1))

logn = np.array(logn, dtype = np.uint8)

transpuesta1 = cv2.transpose(img1)

transpuesta2 = cv2.transpose(img4)

##conca\_verti = cv2.hconcat([img1, img2])

##conca\_hori = cv2.hconcat([suma, resta])

#conca = cv2.vconcat([conca\_verti, conca\_hori])

while True:

    if keyboard.is\_pressed("Y") or keyboard.is\_pressed("y"):

        #cv2.imshow("Operaciones", conca)

        cv2.imshow("image1",img1), cv2.moveWindow("image1", 0, 0)

        cv2.imshow("image2",img4), cv2.moveWindow("image2", 975 , 0)

        cv2.imshow("Suma",suma), cv2.moveWindow("Suma", 500, 250)

        cv2.imshow("Resta", resta), cv2.moveWindow("Resta", 500, 280)

        cv2.imshow("Division", division), cv2.moveWindow("Division", 500, 310)

        cv2.imshow("Multiplicacion", multi), cv2.moveWindow("Multiplicacion", 500, 340)

        cv2.imshow("Potencia 2", potencia), cv2.moveWindow("Potencia 2", 500, 370)

        cv2.imshow("Potencia 4", potencia2), cv2.moveWindow("Potencia 4", 500, 400)

        cv2.imshow("Conjuncion", conjuncion), cv2.moveWindow("Conjuncion", 500, 430)

        cv2.imshow("Disyuncion", disyuncion), cv2.moveWindow("Disyuncion", 500, 460)

        cv2.imshow("Negacion", negacion), cv2.moveWindow("Negacion", 500, 490)

        cv2.imshow("Raiz", raiz), cv2.moveWindow("Raiz", 500, 510)

        cv2.imshow("Traslacion img1", traslacion1), cv2.moveWindow("Traslacion img1", 500, 540)

        cv2.imshow("Traslacion img2", traslacion2), cv2.moveWindow("Traslacion img2", 500, 570)

        cv2.imshow("Rotacion img1", rotacion1), cv2.moveWindow("Rotacion img1", 500, 600)

        cv2.imshow("Rotacion img2", rotacion2), cv2.moveWindow("Rotacion img2", 500, 630)

        cv2.imshow("Escalado", escalado), cv2.moveWindow("Escalado", 500, 660)

        cv2.imshow("Logaritmo Natural 1", logn), cv2.moveWindow("Logaritmo Natural 1", 500, 690)

        ##cv2.imshow("Logaritmo Natural 2", logn2), cv2.moveWindow("Logaritmo Natural 2", 500, 710)

        cv2.imshow("Transpuesta 1", transpuesta1), cv2.moveWindow("Transpuesta 1", 500, 740)

        cv2.imshow("Transpuesta 2", transpuesta2), cv2.moveWindow("Transpuesta 2", 500, 740)

        break

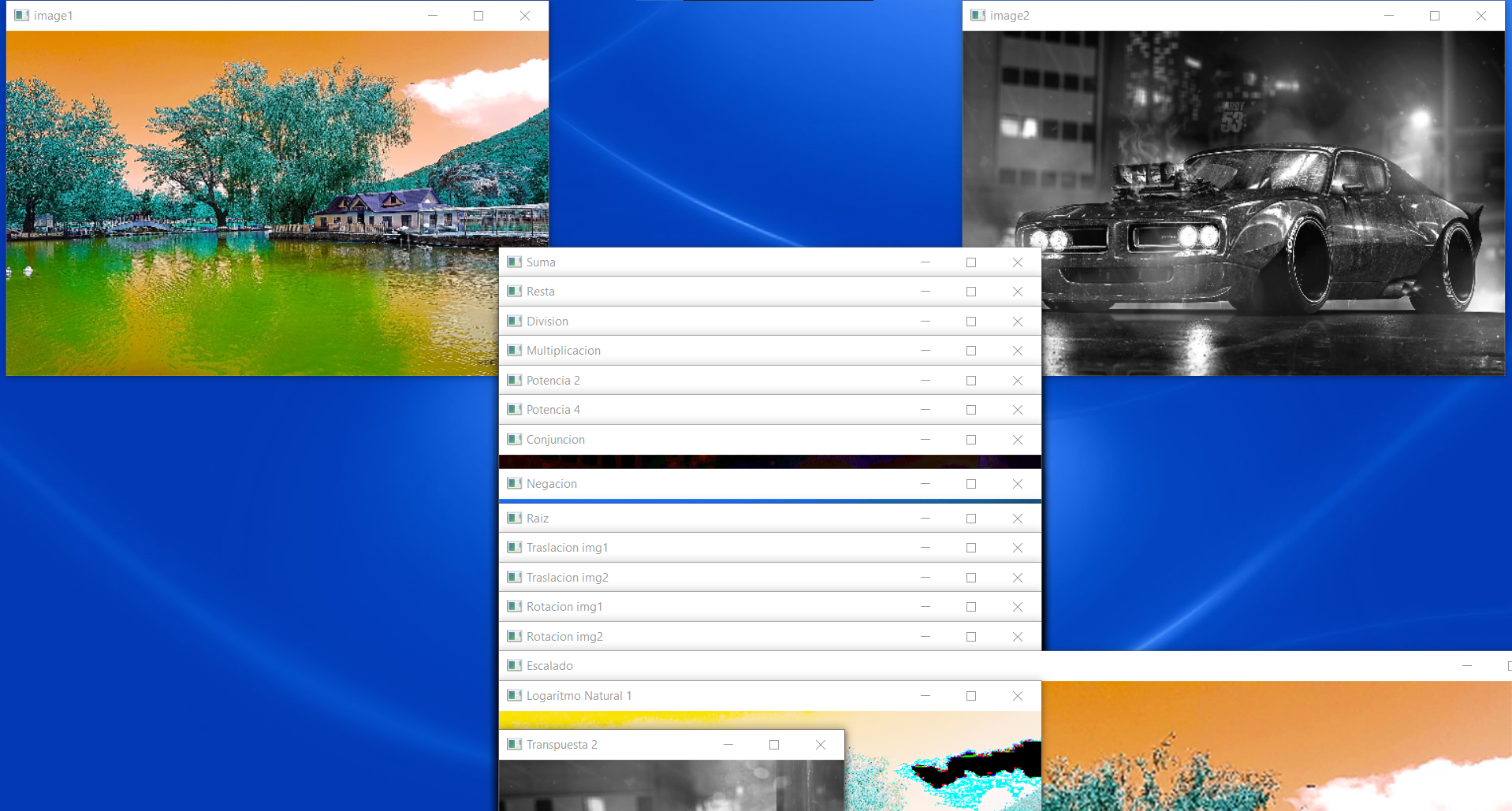
cv2.waitKey(0)

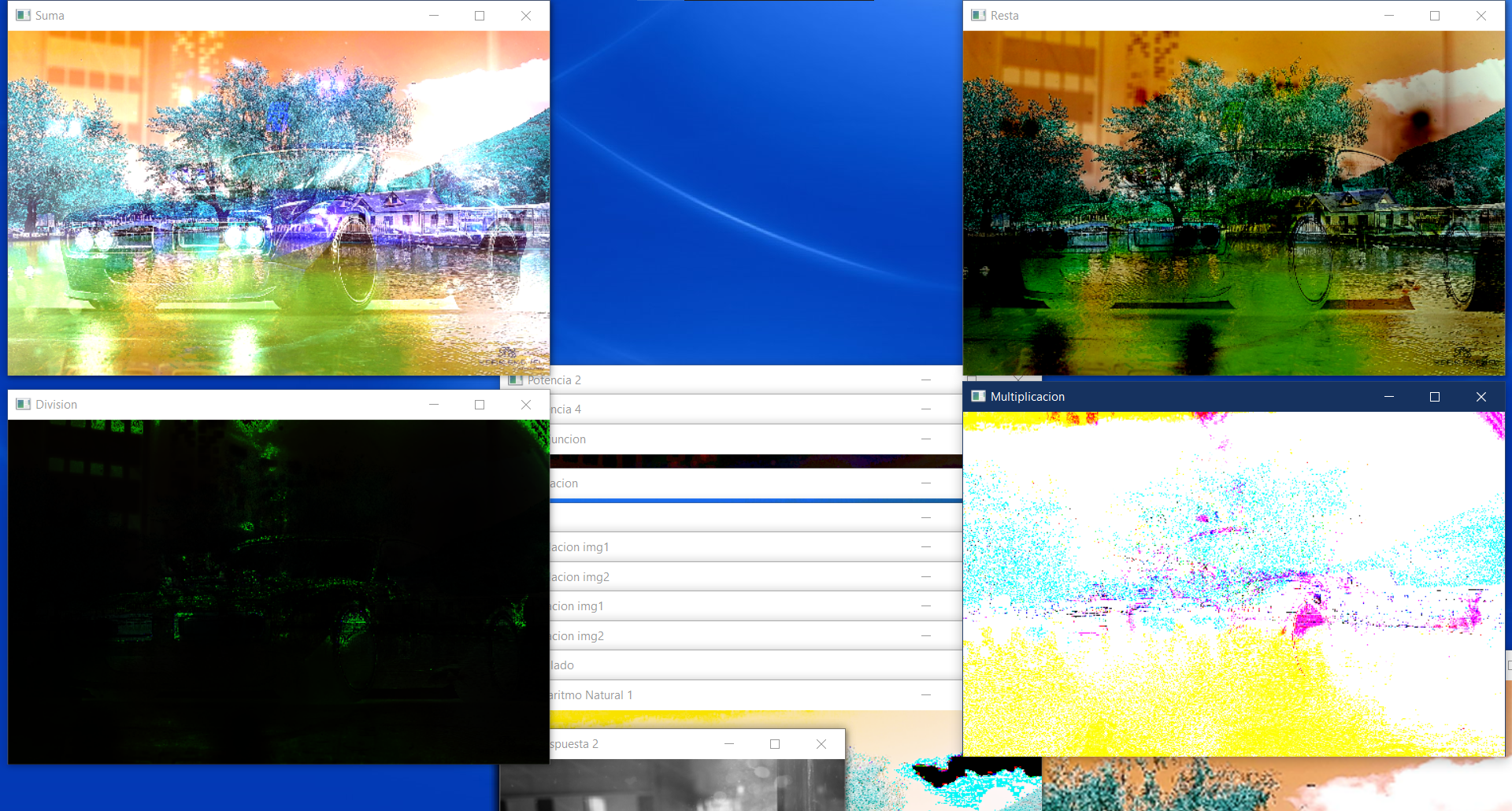
cv2.destroyAllWindows()

**EVIDENCIAS**

Al correr el programa y presionar la tecla inicial de mi nombre, en este caso la letra “Y” nos arroja a la vista las imágenes con las que se trabajaron.

Mostrando así cada una de las operaciones que se hicieron en las imágenes





GITHUB:

<https://github.com/Gylc87/Practica2.git>