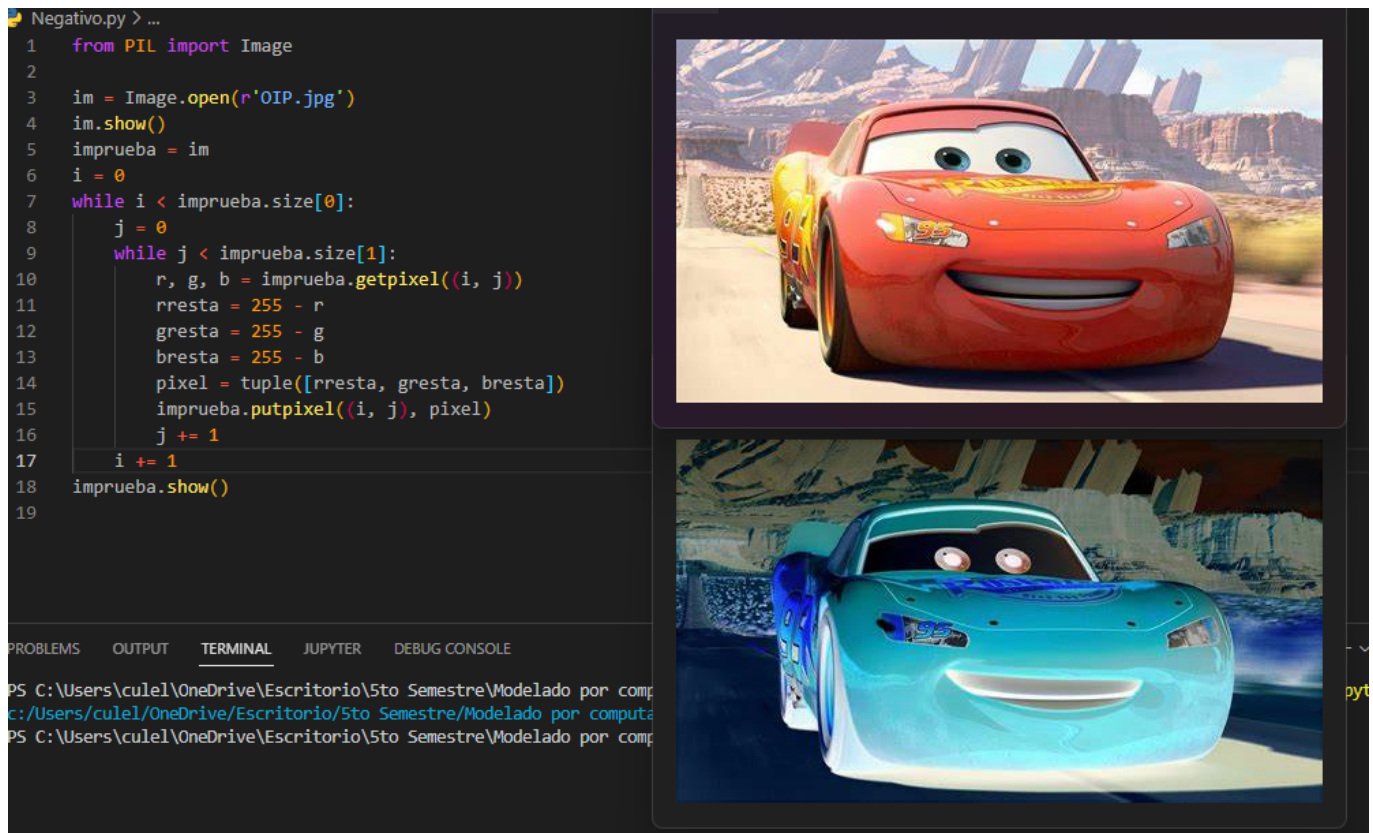


Negativo

```
1  from PIL import Image
2
3  im = Image.open(r'OIP.jpg')
4  im.show()
5  imprueba = im
6  i = 0
7  while i < imprueba.size[0]:
8      j = 0
9      while j < imprueba.size[1]:
10         r, g, b = imprueba.getpixel((i, j))
11         rresta = 255 - r
12         gresta = 255 - g
13         bresta = 255 - b
14         pixel = tuple([rresta, gresta, bresta])
15         imprueba.putpixel((i, j), pixel)
16         j += 1
17     i += 1
18 imprueba.show()
19
```

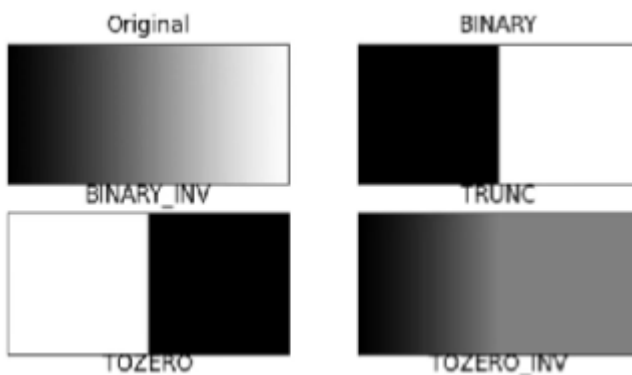
Imagen Original y resultado de Negativa



Método de Umbralización ejemplo

Se muestra los casos de la umbralización

- En imagen original
- Convertida a grises para poder umbralizarla
- Umbralización
- Umbralización Invertida



Umbralización

```
1  from PIL import Image
2  import time
3
4  def Umbralizacion(ggrisBase):
5      tiempoIn = time.time()
6      im = Image.open(r'OIP.jpg')
7      im.show()
8      im7 = im
9      i = 0
10     while i < im7.size[0]:
11         j = 0
12         while j < im7.size[1]:
13             r, g, b = im7.getpixel((i,j))
14             #CONVERTIMOS LA IMAGEN EN GRISES
15             gris = (r + g + b) / 3
16             if gris < ggrisBase:
17                 im7.putpixel((i,j), (0,0,0))
18             else:
19                 im7.putpixel((i,j), (255,255,255))
20             j+=1
21         i+=1
22     im7.show()
23     tiempoFin = time.time()
24     print('El Proceso Tardo: ', tiempoFin - tiempoIn, 'Segundos')
25     im = im.save("Im1probar.jpg")
26     Umbralizacion(100)
```

Imagen umbralizada o binarizada y resultado sobre el tiempo de ejecución del ejercicio

```

5     tiempoIn = time.time()
6     im = Image.open(r'OIP.jpg')
7     im.show()
8     im7 = im
9     i = 0
10    while i < im7.size[0]:
11        j = 0
12        while j < im7.size[1]:
13            r, g, b = im7.getpixel((i,j))
14            #CONVERTIMOS LA IMAGEN EN GRISES
15            gris = (r + g + b) / 3
16            if gris < grisBase:
17                im7.putpixel((i,j), (0,0,0))
18            else:
19                im7.putpixel((i,j), (255,255,255))
20            j+=1
21        i+=1
22    im7.show()
23    tiempoFin = time.time()
24    print('El Proceso Tardo: ', tiempoFin - tiempoIn, 's')
25    im = im.save("Im1probar.jpg")
26    Umbralizacion(100)

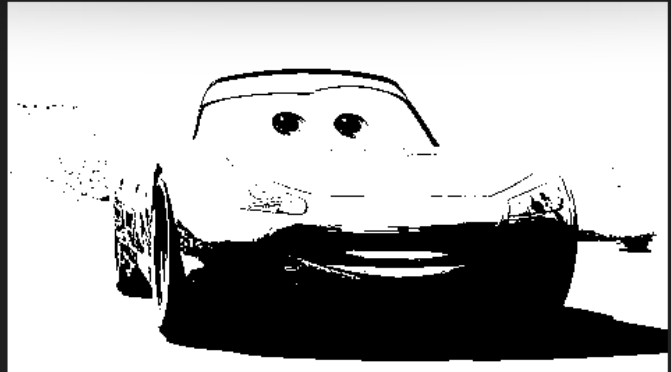
```

PROBLEMS OUTPUT TERMINAL JUPYTER DEBUG CONSOLE

```

PS C:\Users\culel\OneDrive\Escritorio\5to Semestre\Modelado por comp
c:\Users\culel\OneDrive\Escritorio\5to Semestre\Modelado por computa
El Proceso Tardo: 10.284181833267212 Segundos
PS C:\Users\culel\OneDrive\Escritorio\5to Semestre\Modelado por comp

```



Umbralización Inverso

```
1  from PIL import Image
2  import time
3
4  def Umbralizacion(ggrisBase):
5      tiempoIn = time.time()
6      im = Image.open(r'OIP.jpg')
7      im.show()
8      im7 = im
9      i = 0
10     while i < im7.size[0]:
11         j = 0
12         while j < im7.size[1]:
13             r, g, b = im7.getpixel((i,j))
14             #CONVERTIMOS LA IMAGEN EN GRISES
15             gris = (r + g + b) / 3
16             if gris > ggrisBase:
17                 im7.putpixel((i,j), (0,0,0))
18             else:
19                 im7.putpixel((i,j), (255,255,255))
20             j+=1
21         i+=1
22     im7.show()
23     tiempoFin = time.time()
24     print('El Proceso Tardo: ', tiempoFin - tiempoIn, 'Segundos')
25     im = im.save("Im1probar.jpg")
26     Umbralizacion(100)
```

Imagen umbralizada o binarizada y resultado sobre el tiempo de ejecución del ejercicio



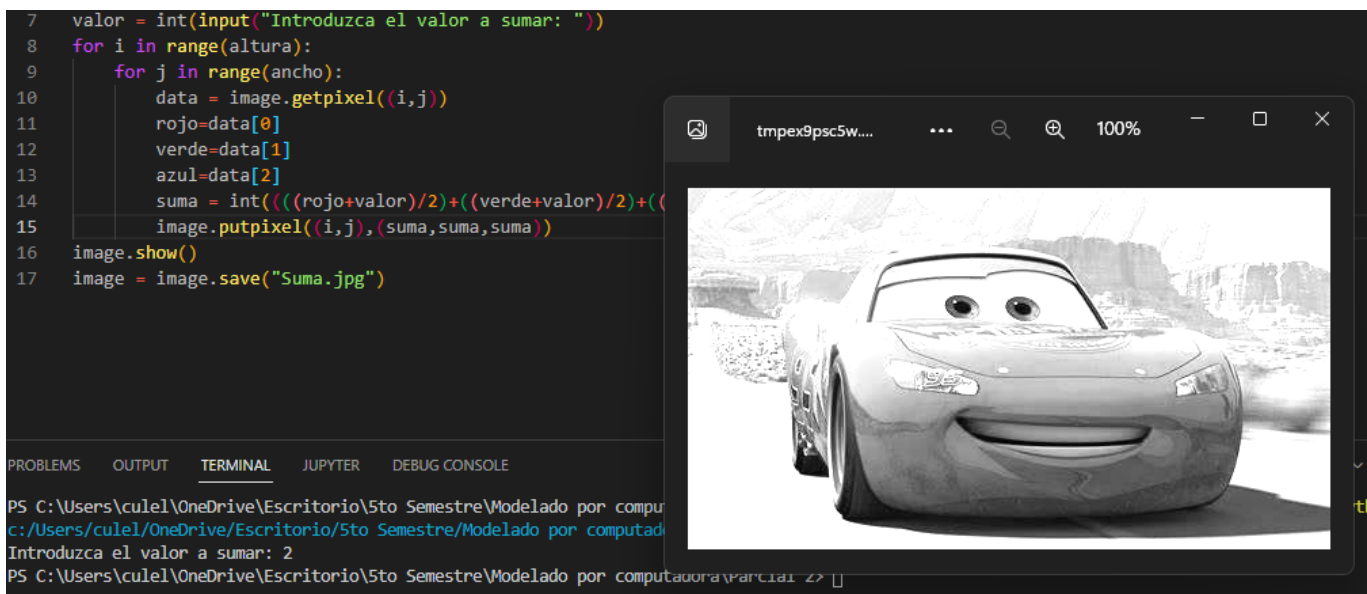
Suma

```

1 from PIL import Image
2
3 image = Image.open(r'OIP.jpg')
4
5 altura = image.size[0]
6 ancho = image.size[1]
7 valor = int(input("Introduzca el valor a sumar: "))
8 for i in range(altura):
9     for j in range(ancho):
10        data = image.getpixel((i,j))
11        rojo=data[0]
12        verde=data[1]
13        azul=data[2]
14        suma = int((((rojo+valor)/2)+((verde+valor)/2)+((azul+valor)/2)))
15        image.putpixel((i,j),(suma,suma,suma))
16 image.show()
17 image = image.save("Suma.jpg")

```

Valor de suma probado con 2



Resta

```
1 from PIL import Image
2
3 image = Image.open(r'OIP.jpg')
4
5 altura = image.size[0]
6 ancho = image.size[1]
7 valor = int(input("Introduzca el valor a restar: "))
8 for i in range(altura):
9     for j in range(ancho):
10         data = image.getpixel((i,j))
11         rojo=data[0]
12         verde=data[1]
13         azul=data[2]
14         resta = int((((rojo-valor)*2)+((verde-valor)*2)+((azul-valor)*2)))
15         image.putpixel((i,j),(resta,resta,resta))
16 image.show()
17 image = image.save("Resta.jpg")
```

Valor de resta probado con 200(oscuro)

```
7 valor = int(input("Introduzca el valor a restar: "))
8 for i in range(altura):
9     for j in range(ancho):
10         data = image.getpixel((i,j))
11         rojo=data[0]
12         verde=data[1]
13         azul=data[2]
14         resta = int((((rojo-valor)*2)+((verde-valor)*2)+(
15             image.putpixel((i,j),(resta,resta,resta))
16 image.show()
17 image = image.save("Resta.jpg")
```

PROBLEMS OUTPUT **TERMINAL** JUPYTER DEBUG CONSOLE

Copyright (C) Microsoft Corporation. Todos los derechos reservados.

Instale la versión más reciente de PowerShell para obtener nuevas car

PS C:\Users\culel\OneDrive\Escritorio\5to Semestre\Modelado por computadora\Parcial 2> & C:/Users/culel/AppData/Local/Microsoft/WindowsApps/python3.9.0/python.exe c:/Users/culel/OneDrive/Escritorio/5to Semestre/Modelado por computadora/Parcial 2/Resta.py

Introduzca el valor a restar: 200

PS C:\Users\culel\OneDrive\Escritorio\5to Semestre\Modelado por computadora\Parcial 2> █

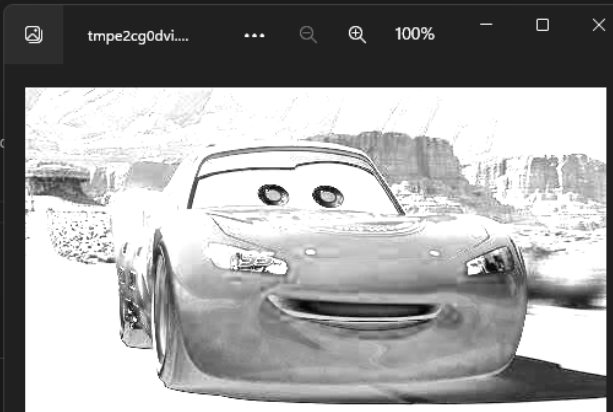


Resta con valor absoluto

```
1 from PIL import Image
2
3 image = Image.open(r'OIP.jpg')
4
5 altura = image.size[0]
6 ancho = image.size[1]
7 valor = int(input("Introduzca el valor a restar: "))
8 for i in range(altura):
9     for j in range(ancho):
10         data = image.getpixel((i,j))
11         rojo=data[0]
12         verde=data[1]
13         azul=data[2]
14         resta = int((((abs(rojo-valor))+abs(verde-valor))+abs(azul-valor))))
15         image.putpixel((i,j),(resta,resta,resta))
16 image.show()
17 image = image.save("Resta.jpg")
```

Valor de resta probado con 80


```
7 valor = int(input("Introduzca el valor a restar: "))
8 for i in range(altura):
9     for j in range(ancho):
10         data = image.getpixel((i,j))
11         rojo=data[0]
12         verde=data[1]
13         azul=data[2]
14         resta = int(((abs(rojo-valor))+abs(verde-valor))+abs(azul-valor))
15         image.putpixel((i,j),(resta,resta,resta))
16 image.show()
17 image = image.save("Resta.jpg")
```



Introduzca el valor a restar: 20
PS C:\Users\culel\OneDrive\Escritorio\5to Semestre\Modelado por computadora\Parcial 2> & C:/Users/culel/AppData/Local/Microsoft/WindowsApps/python3.10.exe "c:/Users/culel/OneDrive/Escritorio/5to Semestre/Modelado por computadora/Parcial 2/Resta.py"
Introduzca el valor a restar: 200
PS C:\Users\culel\OneDrive\Escritorio\5to Semestre\Modelado por computadora\Parcial 2> & C:/Users/culel/AppData/Local/Microsoft/WindowsApps/python3.10.exe "c:/Users/culel/OneDrive/Escritorio/5to Semestre/Modelado por computadora/Parcial 2/Resta.py"
Introduzca el valor a restar: 80
PS C:\Users\culel\OneDrive\Escritorio\5to Semestre\Modelado por computadora\Parcial 2> |

Multiplicación

```
1 from PIL import Image
2
3 image = Image.open(r'OIP.jpg')
4
5 altura = image.size[0]
6 ancho = image.size[1]
7 valor = int(input("Introduzca el valor a multiplicar: "))
8 for i in range(altura):
9     for j in range(ancho):
10         data = image.getpixel((i,j))
11         rojo=data[0]
12         verde=data[1]
13         azul=data[2]
14         multiplicacion = int((((rojo*valor)/2)+((verde*valor)/2)+((azul*valor)/2)))
15         image.putpixel((i,j),(multiplicacion,multiplicacion,multiplicacion))
16 image.show()
17 image = image.save("Multi.jpg")
```

Valor de multiplicación probado con 5

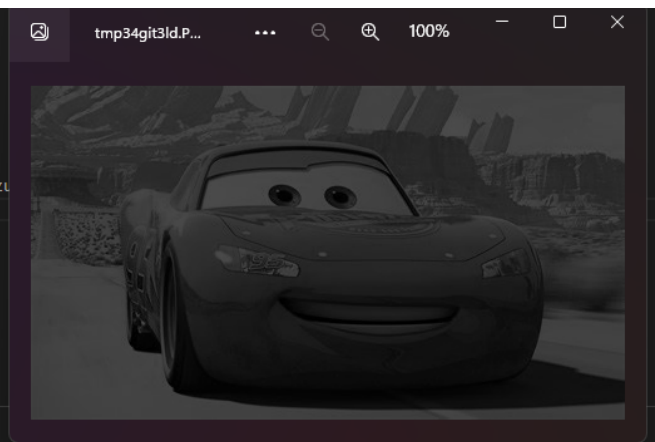


División

```
1 from PIL import Image
2
3 image = Image.open(r'OIP.jpg')
4
5 altura = image.size[0]
6 ancho = image.size[1]
7 valor = int(input("Introduzca el valor a Dividir: "))
8 for i in range(altura):
9     for j in range(ancho):
10         data = image.getpixel((i,j))
11         rojo=data[0]
12         verde=data[1]
13         azul=data[2]
14         dividir = int((((abs(rojo/valor))+abs(verde/valor))+abs(azul/valor))))
15         image.putpixel((i,j),(dividir,dividir,dividir))
16 image.show()
17 image = image.save("Divison.jpg")
```

Valor de división probado con 5

```
7 valor = int(input("Introduzca el valor a Dividir: "))
8 for i in range(altura):
9     for j in range(ancho):
10         data = image.getpixel((i,j))
11         rojo=data[0]
12         verde=data[1]
13         azul=data[2]
14         dividir = int(((abs(rojo/valor))+abs(verde/valor))+abs(azu
15         image.putpixel((i,j),(dividir,dividir,dividir))
16 image.show()
17 image = image.save("Divison.jpg")
```



PROBLEMS OUTPUT TERMINAL JUPYTER DEBUG CONSOLE

```
PS C:\Users\culel\OneDrive\Escritorio\5to Semestre\Modelado por computadora\Parcial 2> & C:/Users/culel/AppData/Local/Microsoft/WindowsApps/python3.10.exe
c:/Users/culel/OneDrive/Escritorio/5to Semestre/Modelado por computadora/Parcial 2/Division.py"
```

```
Introduzca el valor a Dividir: 10
```

```
PS C:\Users\culel\OneDrive\Escritorio\5to Semestre\Modelado por computadora\Parcial 2> █
```