

Practica 4: filtro paleta de colores.

```
import os

# 1. Asegurar que la carpeta exista
if not os.path.exists('./images'):
    os.makedirs('./images')

# 2. Abrir los archivos (Asegúrate de haber subido volcan.bmp a la carpeta images)
# Usamos 'volcan.bmp' como base para aplicar el filtro
try:
    file = open('./images/volcan.bmp', 'rb')
    fileo = open('./images/volcan_filtrado.bmp', 'wb')

    # Leer cabecera (54 bytes) y copiarla al archivo de salida
    metadata = file.read(54)
    fileo.write(metadata)

    # 3. Definición de los 18 tonos (BGR para BMP)
    # IMPORTANTE: Aquí debes poner los colores que extrajiste de 'morado.bmp'
    # El formato es [Azul, Verde, Rojo] en hexadecimal
    t = [
        [0x11, 0x00, 0x11], [0x22, 0x00, 0x22], [0x33, 0x00, 0x33],
        [0x44, 0x00, 0x44], [0x55, 0x00, 0x55], [0x66, 0x00, 0x66],
        [0x77, 0x00, 0x77], [0x88, 0x00, 0x88], [0x99, 0x00, 0x99],
        [0xAA, 0x00, 0xAA], [0xBB, 0x00, 0xBB], [0xCC, 0x00, 0xCC],
        [0xDD, 0x00, 0xDD], [0xEE, 0x00, 0xEE], [0xFF, 0x00, 0xFF],
        [0xFF, 0x33, 0xFF], [0xFF, 0x66, 0xFF], [0xFF, 0x99, 0xFF]
    ]

    # 4. Configuración de la cuantización (18 niveles)
    file.seek(54, 0)
    no_pix = 0
    # Dividimos el rango total de 24 bits entre 18
    salto = (pow(2, 24) - 1) // 18

    print("Procesando imagen...")

    while True:
        pixel_data = file.read(3)
        if len(pixel_data) > 0:
            # Convertir los 3 bytes del píxel a un entero para comparar
            valor_int = int.from_bytes(bytes(pixel_data), byteorder='little')

            # 5. Aplicación de los 17 límites para los 18 tonos
            if valor_int < salto:
                fileo.write(bytes(t[0]))
            elif valor_int < salto * 2:
```

```

        fileo.write(bytes(t[1]))
    elif valor_int < salto * 3:
        fileo.write(bytes(t[2]))
    elif valor_int < salto * 4:
        fileo.write(bytes(t[3]))
    elif valor_int < salto * 5:
        fileo.write(bytes(t[4]))
    elif valor_int < salto * 6:
        fileo.write(bytes(t[5]))
    elif valor_int < salto * 7:
        fileo.write(bytes(t[6]))
    elif valor_int < salto * 8:
        fileo.write(bytes(t[7]))
    elif valor_int < salto * 9:
        fileo.write(bytes(t[8]))
    elif valor_int < salto * 10:
        fileo.write(bytes(t[9]))
    elif valor_int < salto * 11:
        fileo.write(bytes(t[10]))
    elif valor_int < salto * 12:
        fileo.write(bytes(t[11]))
    elif valor_int < salto * 13:
        fileo.write(bytes(t[12]))
    elif valor_int < salto * 14:
        fileo.write(bytes(t[13]))
    elif valor_int < salto * 15:
        fileo.write(bytes(t[14]))
    elif valor_int < salto * 16:
        fileo.write(bytes(t[15]))
    elif valor_int < salto * 17:
        fileo.write(bytes(t[16]))
    else:
        fileo.write(bytes(t[17]))

    no_pix += 1
else:
    break

```

```

file.close()
fileo.close()
print(f'¡Listo! Píxeles procesados: {no_pix}')
print('Busca el archivo "volcan_filtrado.bmp" en la carpeta images.')

```

```

except FileNotFoundError:
    print("Error: No se encontró 'volcan.bmp' en la carpeta images. ¡Cárgalo primero!")

```

