**From PIL import Image**

Import os

Import random

Def encrypt\_image(input\_image\_path, output\_image\_path, key):

Original\_image = Image.open(input\_image\_path)

Pixels = original\_image.load()

Width, height = original\_image.size

For I in range(width):

For j in range(height):

If I % 2 == 0 and j % 2 == 0:

# Swap pixel values if the sum of the pixel’s RGB values is even

If sum(pixels[I, j]) % 2 == 0:

Temp = pixels[I, j]

Pixels[I, j] = pixels[I + key, j + key]

Pixels[I + key, j + key] = temp

Original\_image.save(output\_image\_path)

Def decrypt\_image(input\_image\_path, output\_image\_path, key):

Encrypted\_image = Image.open(input\_image\_path)

Pixels = encrypted\_image.load()

Width, height = encrypted\_image.size

For I in range(width):

For j in range(height):

If I % 2 == 0 and j % 2 == 0:

# Swap pixel values if the sum of the pixel’s RGB values is even

If sum(pixels[I, j]) % 2 == 0:

Temp = pixels[I, j]

Pixels[I, j] = pixels[I + key, j + key]

Pixels[I + key, j + key] = temp

Encrypted\_image.save(output\_image\_path)

Def main():

Input\_image\_path = “input.png”

Output\_image\_path = “output.png”

Key = 5

Print(“Welcome to the Image Encryption Tool!”)

Choice = input(“Would you like to €ncrypt or (d)ecrypt an image? “).lower()

If choice not in [‘e’, ‘d’]:

Print(“Invalid choice, please enter ‘e’ for encrypt or ‘d’ for decrypt.”)

Return

If choice == ‘e’:

Print(f”Encrypting image ‘{input\_image\_path}’…”)

Encrypt\_image(input\_image\_path, output\_image\_path, key)

Print(f”Encrypted image saved as ‘{output\_image\_path}’.”)

Elif choice == ‘d’:

Print(f”Decrypting image ‘{input\_image\_path}’…”)

Decrypt\_image(input\_image\_path, output\_image\_path, key)

Print(f”Decrypted image saved as ‘{output\_image\_path}’.”)

If \_\_name\_\_ == “\_\_main\_\_”:

Main()