PROJECT REPORT

Encryption and Decryption Between Users in Morse Using a Flashlight

This project performs text communication using Morse code by blinking a simulated flashlight (screen background). It consists of three parts: Encryption (Text to Morse), Transmission (Flashing Light), and Decryption (Morse to Text). No external hardware components are required.

Part 1: Encryption (Text → Morse)

```
MORSE_CODE_DICT = {
    'A': '.-', 'B': '-...', 'C': '-.-.', 'D': '-..',
    'E': '.', 'F': '..-.', 'G': '--.', 'H': '....',
    'I': '..', 'J': '.---', 'K': '-.-', 'L': '.-..',
    'M': '--', 'N': '-.', 'O': '---', 'P': '.--.',
    'Q': '--.-', 'R': '.-.', 'S': '...', 'T': '-',
    'U': '..-', 'V': '...-', 'W': '.--', 'X': '-..-',
    'Y': '-.--', 'Z': '--..', '1': '.----', '2': '..---',
    '3': '...--', '4': '....-', '5': '.....', '6': '-....',
    '7': '--...', '8': '---..', '9': '----.', '0': '-----',
    ', ': '--..--', '.': '.-.-.-', '?': '..--..', '/': '-..-.',
    '-': '-....-', '(': '-.--.', ')': '-.--.-', ' ': '/'
}

def text_to_morse(text):
    text = text.upper()
    morse_code = ' '.join(MORSE_CODE_DICT.get(char, '') for char in text)
    return morse_code

if __name__ == "__main__":
    msg = input("Enter text to encrypt: ")
    morse = text_to_morse(msg)
    print("Encrypted Morse Code:", morse)
```

Part 2: Flashlight Transmission (Morse → Blinking Light)

```
import tkinter as tk
import time
from morse_encrypt import text_to_morse

DOT_TIME = 0.2  # seconds for a dot
DASH_TIME = DOT_TIME * 3
GAP = DOT_TIME

def blink_flashlight(morse_code):
    root = tk.Tk()
    root.title("Flashlight Transmission")
    root.geometry("400x400")
    frame = tk.Frame(root, bg="black")
    frame.pack(fill="both", expand=True)

    def flash_on():
        frame.config(bg="white")
        root.update()

    def flash_off():
        frame.config(bg="black")
        root.update()

    for symbol in morse_code:
```

```
        if symbol == '.':
            flash_on()
            time.sleep(DOT_TIME)
            flash_off()
        elif symbol == '-':
            flash_on()
            time.sleep(DASH_TIME)
            flash_off()
        elif symbol == ' ':
            time.sleep(DOT_TIME * 3)  # gap between letters
        elif symbol == '/':
            time.sleep(DOT_TIME * 7)  # gap between words
        time.sleep(GAP)
    root.destroy()

if __name__ == "__main__":
    msg = input("Enter text to send: ")
    morse = text_to_morse(msg)
    print("Morse Code:", morse)
    blink_flashlight(morse)
```

Part 3: Decryption (Morse → Text)

```
REVERSE_MORSE_DICT = {
    '.-': 'A', '-...': 'B', '-.-.': 'C', '-..': 'D',
    '.': 'E', '..-.': 'F', '--.': 'G', '....': 'H',
    '..': 'I', '.---': 'J', '-.-': 'K', '.-..': 'L',
    '--': 'M', '-.': 'N', '---': 'O', '.--.': 'P',
    '--.-': 'Q', '.-.': 'R', '...': 'S', '-': 'T',
    '..-': 'U', '...-': 'V', '.--': 'W', '-..-': 'X',
    '-.--': 'Y', '--..': 'Z', '-----': '0', '.----': '1',
    '..---': '2', '...--': '3', '....-': '4', '.....': '5',
    '-....': '6', '--...': '7', '---..': '8', '----.': '9',
    '/': ' '
}

def morse_to_text(morse_code):
    words = morse_code.split(' / ')
    decoded_message = ''
    for word in words:
        for symbol in word.split():
            decoded_message += REVERSE_MORSE_DICT.get(symbol, '')
        decoded_message += ' '
    return decoded_message.strip()

if __name__ == "__main__":
    morse_input = input("Enter received Morse code (use / for space): ")
    text = morse_to_text(morse_input)
    print("Decrypted Text:", text)
```

Sample Output

```
Input:
HELLO WORLD

Encrypted Morse Code:
.... . .-.. .-.. --- / .-- --- .-. .-.. -..

Flashlight Transmission:
■ Blinks for dots and dashes according to Morse timing.

Decrypted Output:
HELLO WORLD
```